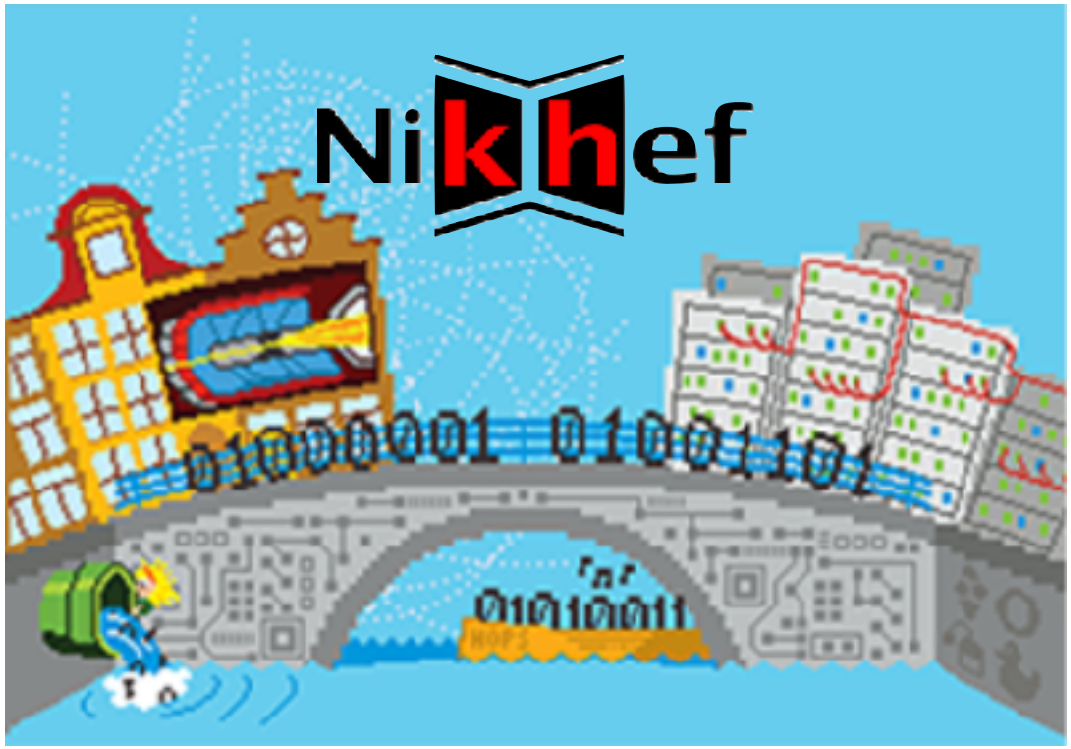
The TDAQ System of the MEG Experiment and its upgrade

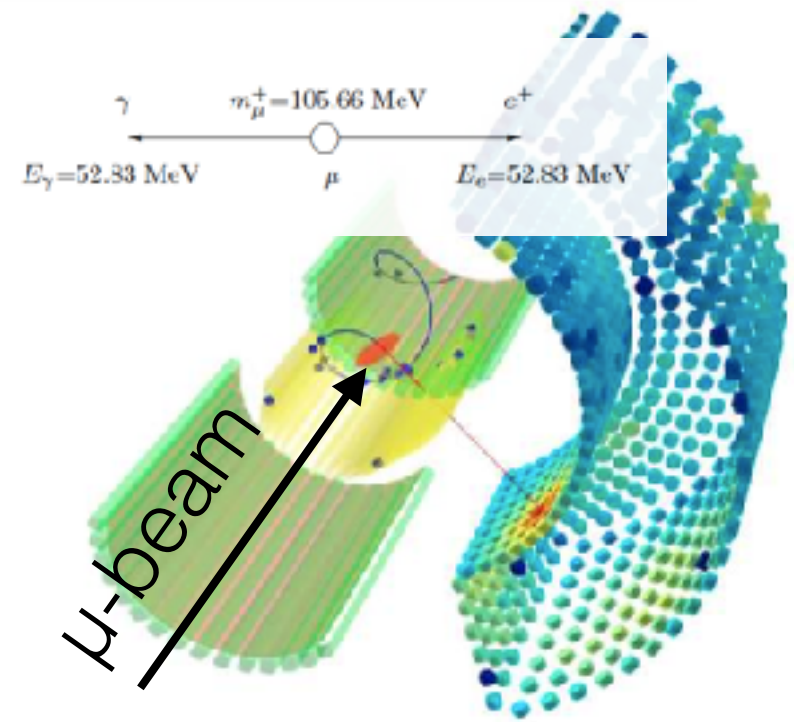
ISOTDAQ 2017

- The MEG experiment
- requirements for the TDAQ from physics
  - DAQ choice
  - requirements for the trigger
- An FPGA based trigger
  - algorithm implementation
- TDAQ efficiency considerations
- The MEG II case
  - upgrade design
  - the upgraded TDAQ system



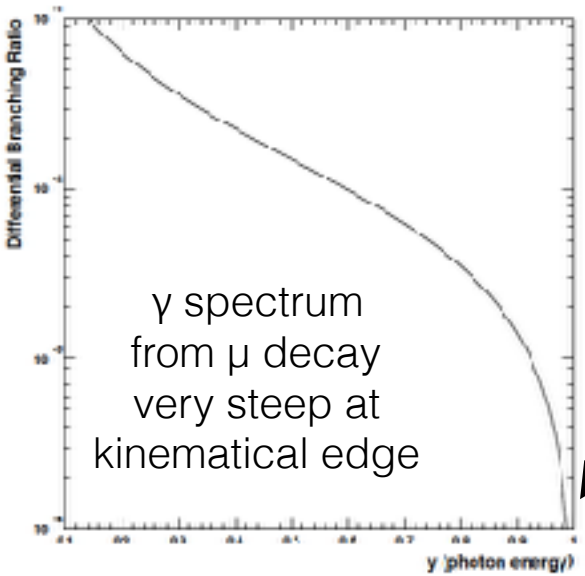
Luca Galli, INFN Sezione di Pisa
luca.galli@pi.infn.it

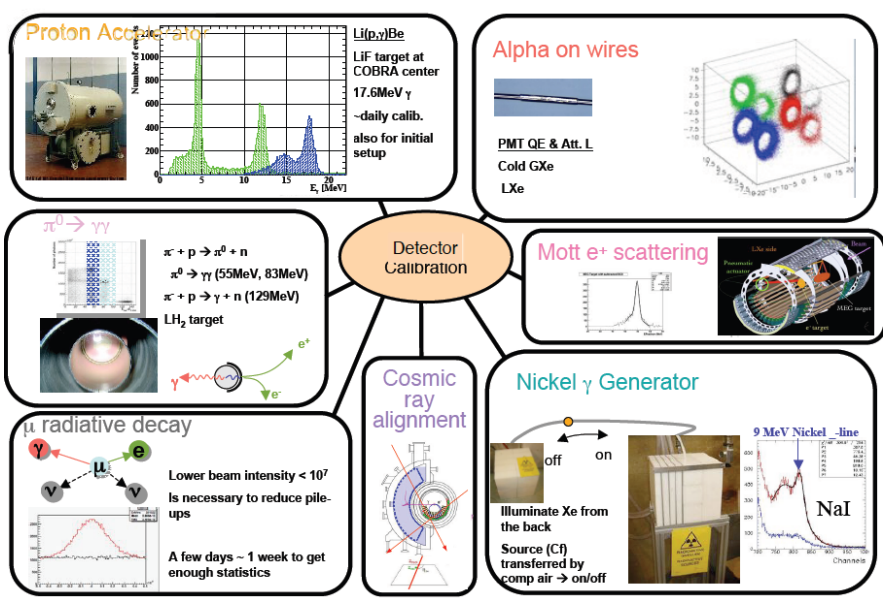
Very few words about the MEG experiment



- "Tiny" collaboration
- Search $\mu \to e\gamma$ with 5x10^(-13) sensitivity on the process @Paul Scherrer Institut, Villigen Switzerland
  - *CERN recognised experiment in the intensity frontier*
  - *prohibited in SM -> new physics?/! (SUSY?)*
- positive μ-beam stopped in a thin target (3x10^7 μ/sec)
  - *positron detector*
    - non-uniform magnetic field COBRA to bend positrons
    - tracking with segmented wire drift chambers
    - timing with plastic scintillator bars read by PMTs (Timing Counter)
  - *photon detector*
    - Liquid Xenon calorimeter read with PMTs

Requirements to the TDAQ

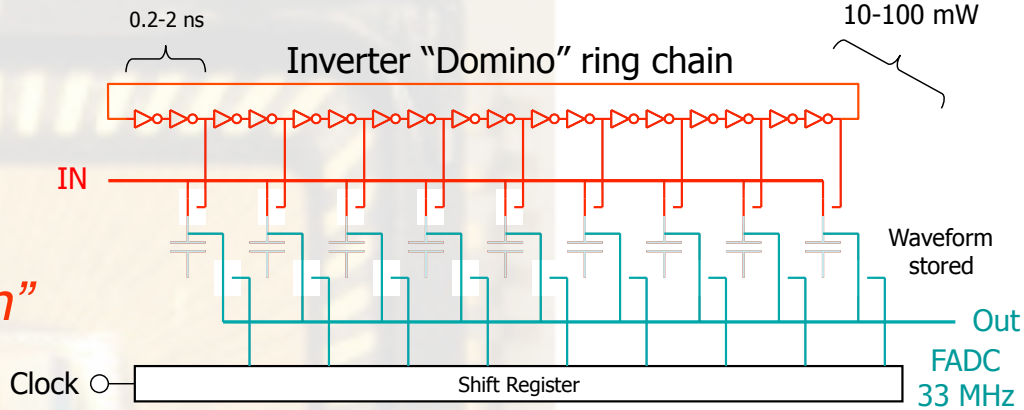


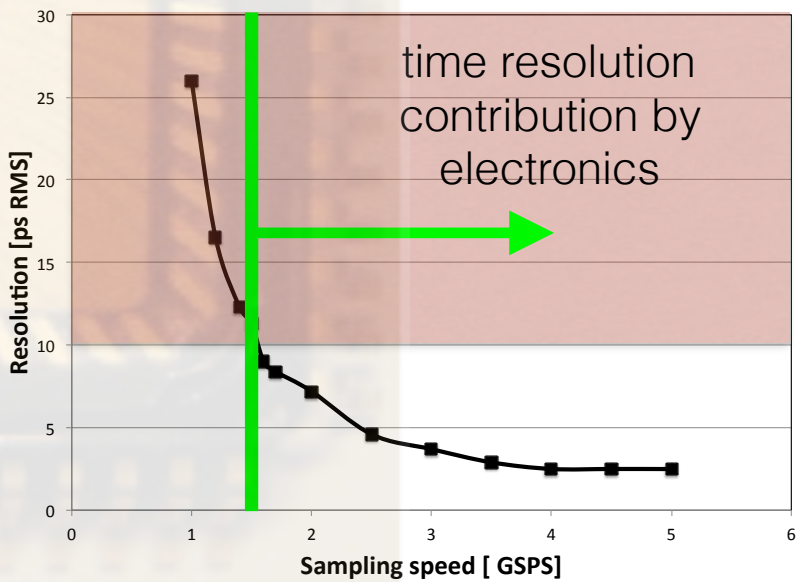


- An experiment in the intensity frontier demands for:
  - *statistics -> high beam intensity (and associated raw event rate and detector occupancy)*
    - electronics to resolve any possible pile-up within few ns in detectors
      - *use of GSPS waveform digitisers, no TDC and QDC*
  - *resolution -> background suppression*
    - development of detector with unprecedented resolutions at signal energy (52.8 MeV)
      - *high precision electronics for charge and time measurements*
  - *stability -> systematics under control*
    - the detector stability measured with a redundant set of calibration methods (evolving year by year)
      - *trigger system flexible to cope with any experimental requests*

Readout choice: the DRS chip

- The Domino Ring Sampler
  - *waveform digitiser developed at Paul Scherrer Institut*
  - *capacitor array to store the charge from detector signals, each capacitor is a "bin" in our waveform*
    - read out with external ADC
  - *sampling speed tuneable from 800 MSPS to 5 GSPS*
  - *waveform width = 1024 bins (from 12.8 μs to 200 ns)*
    - time resolution demands for sampling speed greater that 1.5 GSPS - 600 ns memory depth
      - *requirement on trigger latency!*

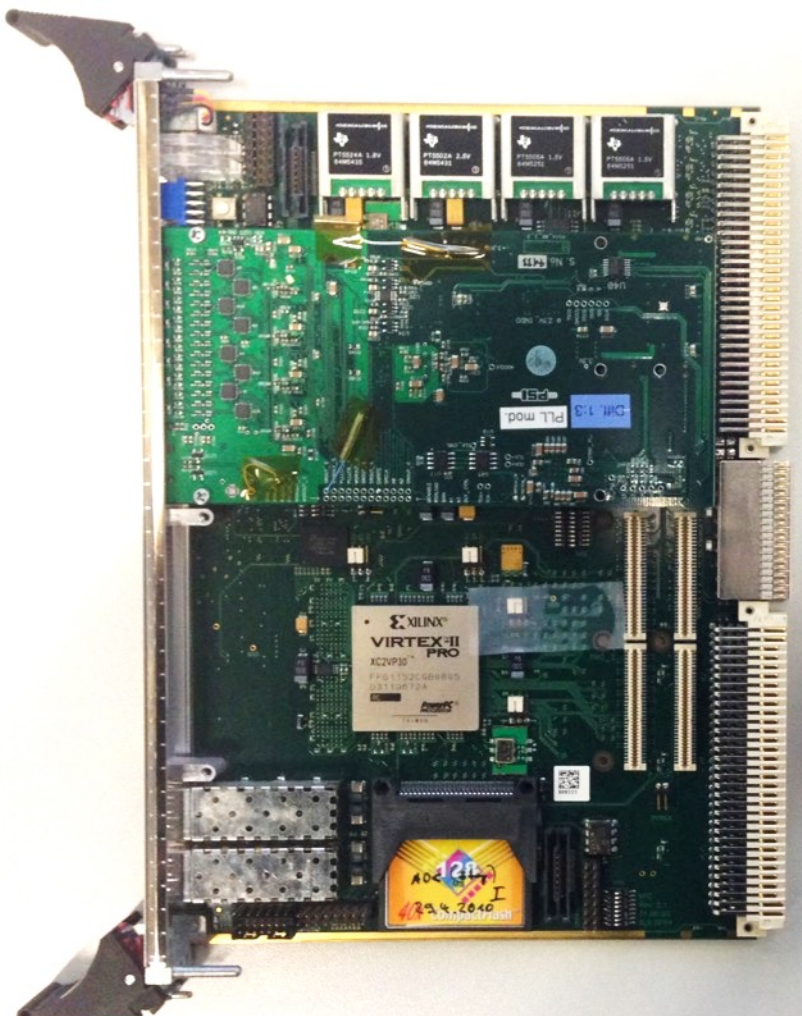


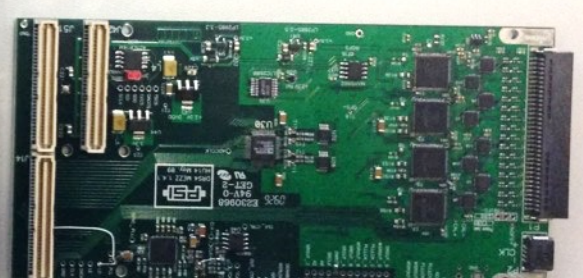


DAQ electronics

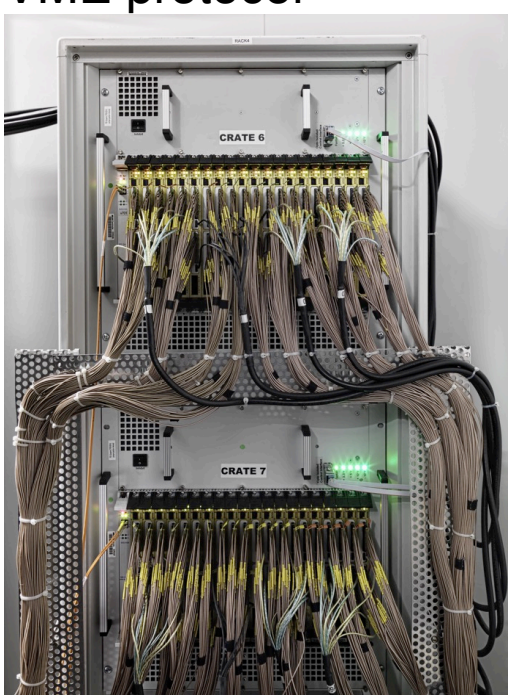
- DRS chips in VME boards
  - *32 channel per board*
- 5 crates
  - *20 boards per crate*
    - 640 channels per crate

Introduction to VME bus
M. Joos

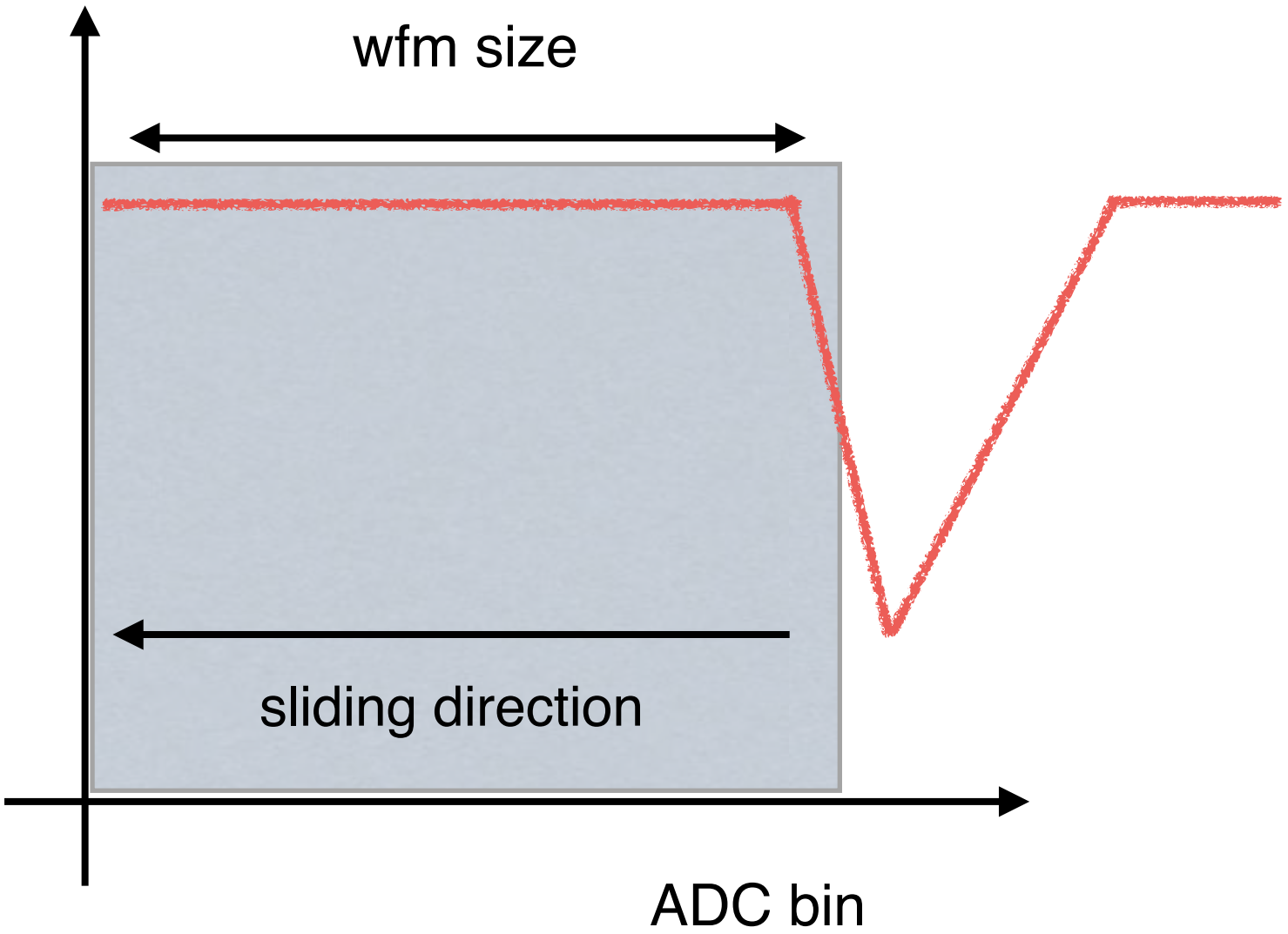




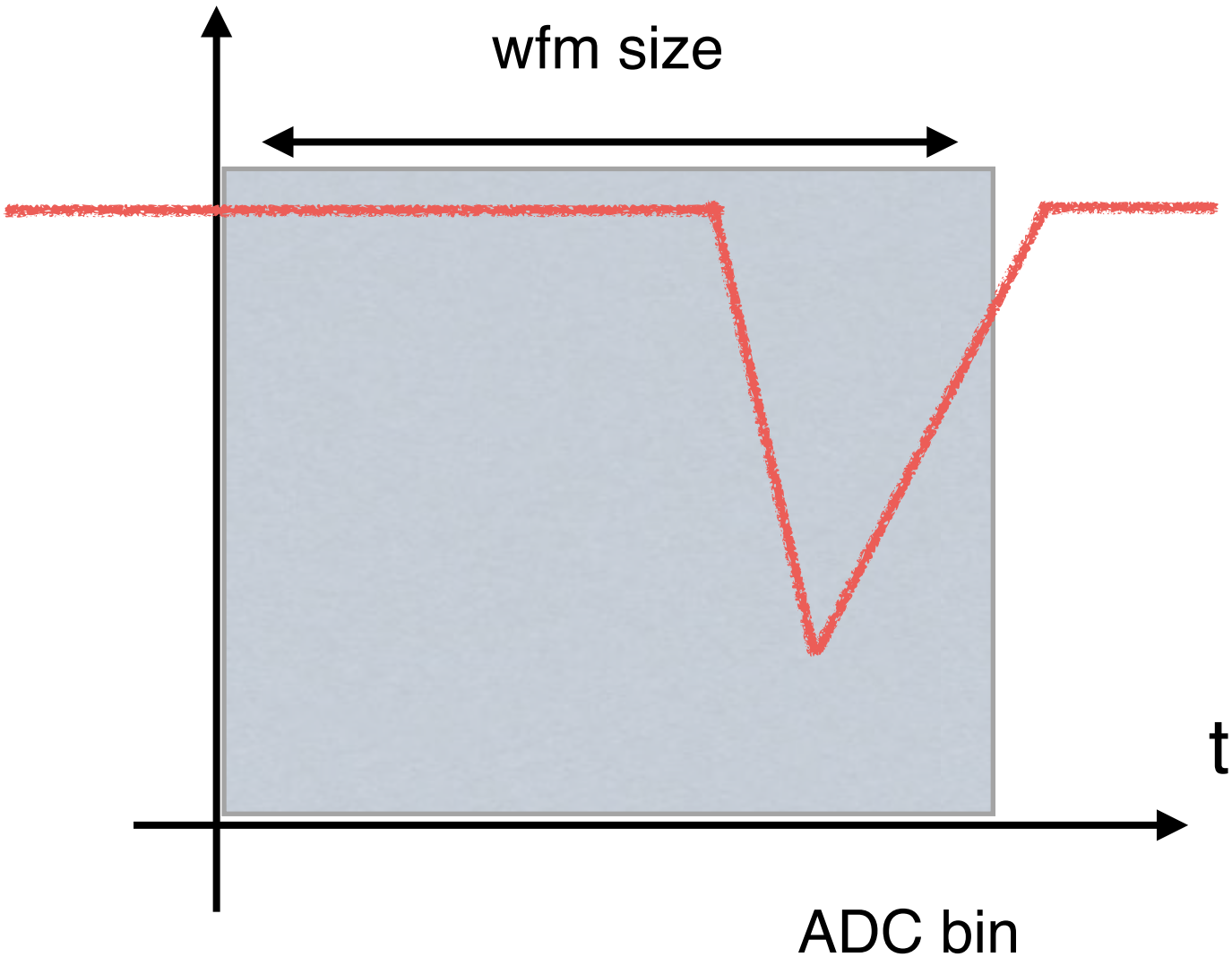
- Boards read-out with 32 bit - 2eVME protocol
  - *80 MB/s transfer speed*



Event occurence



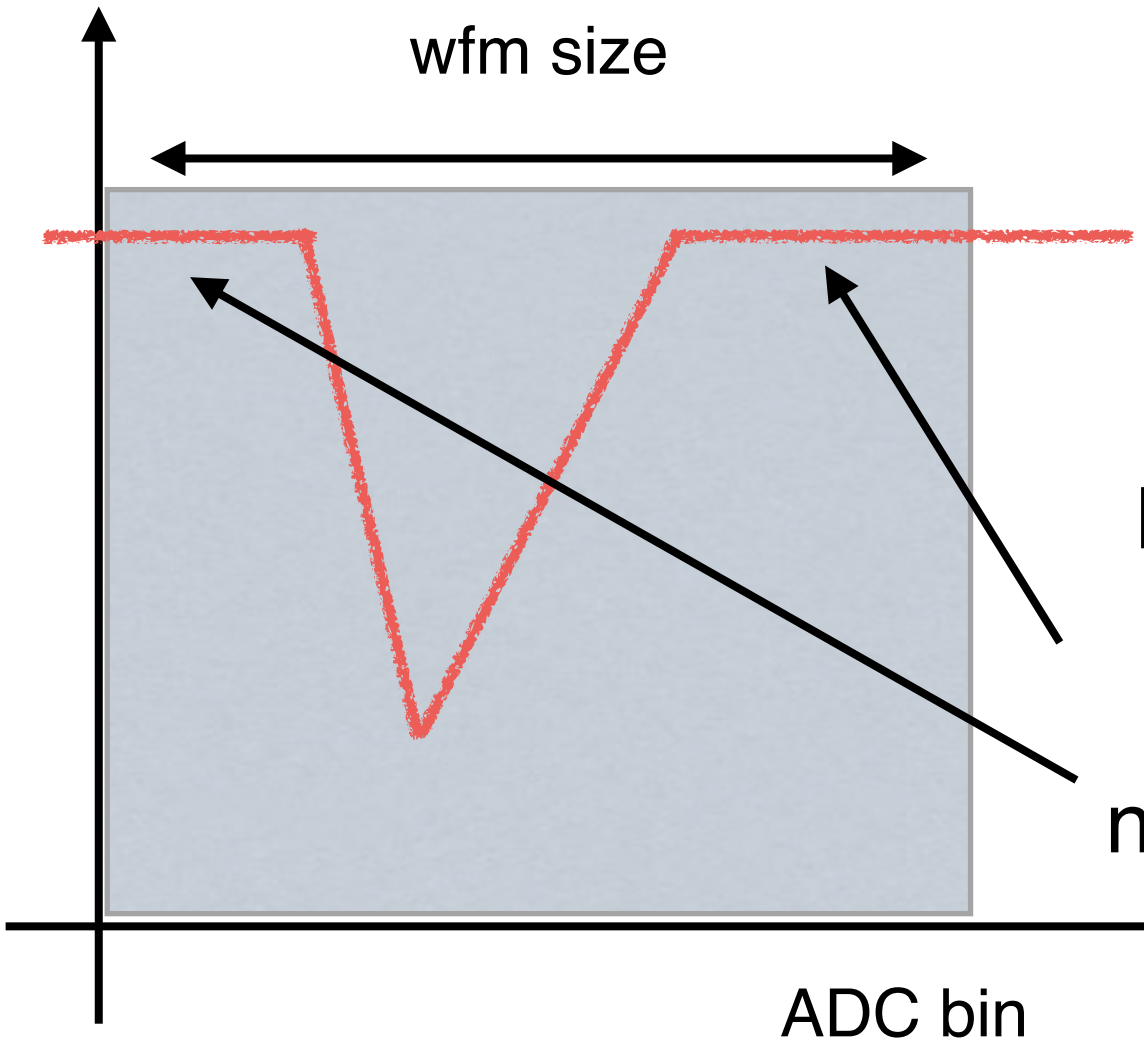
Trigger too early



Signal trailing edge partially lost

Don’t worry, usually the trigger is never too fast….

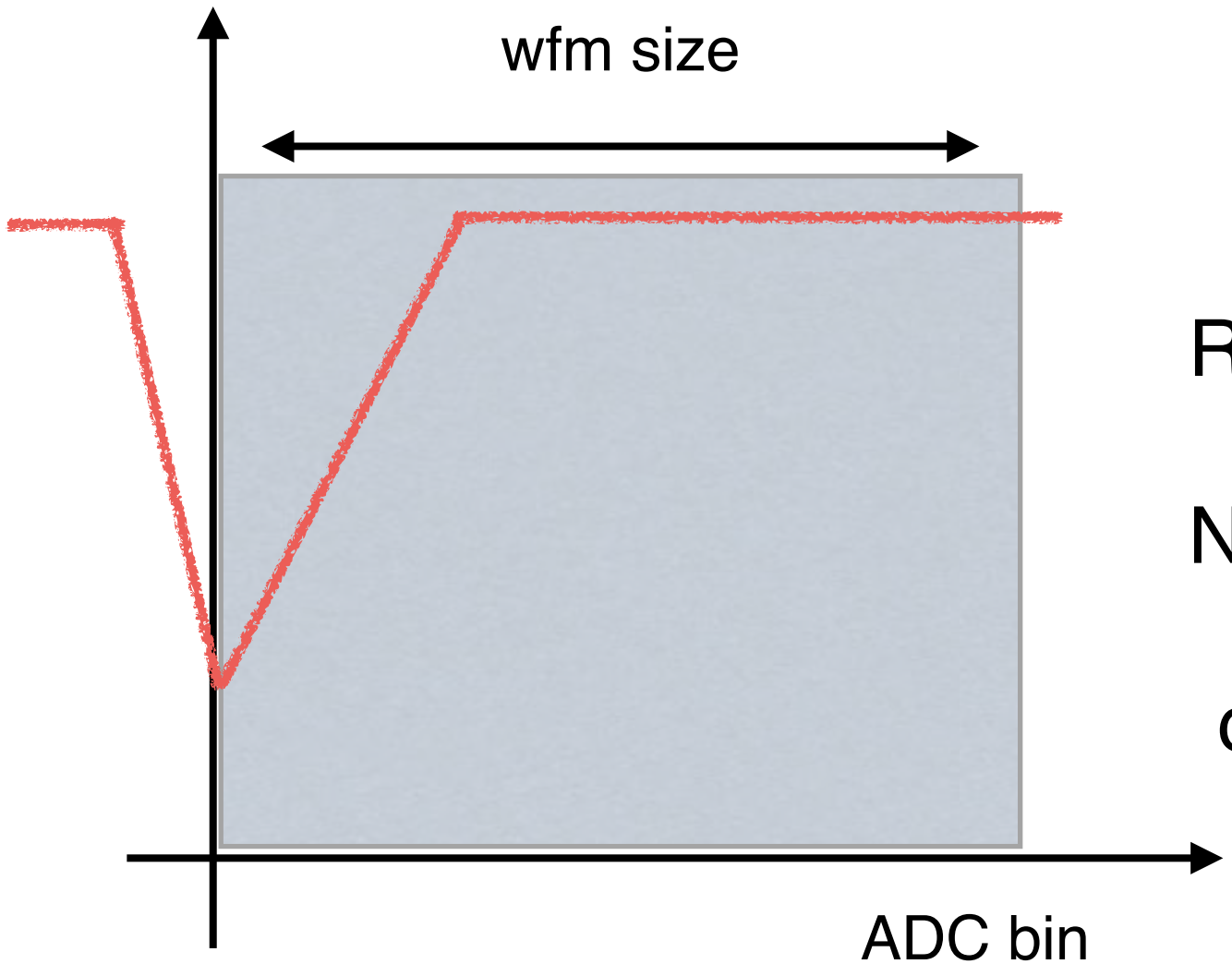
Right trigger time



Wfm 100% contained

Baseline before and after the pulse to check noise and pile-up

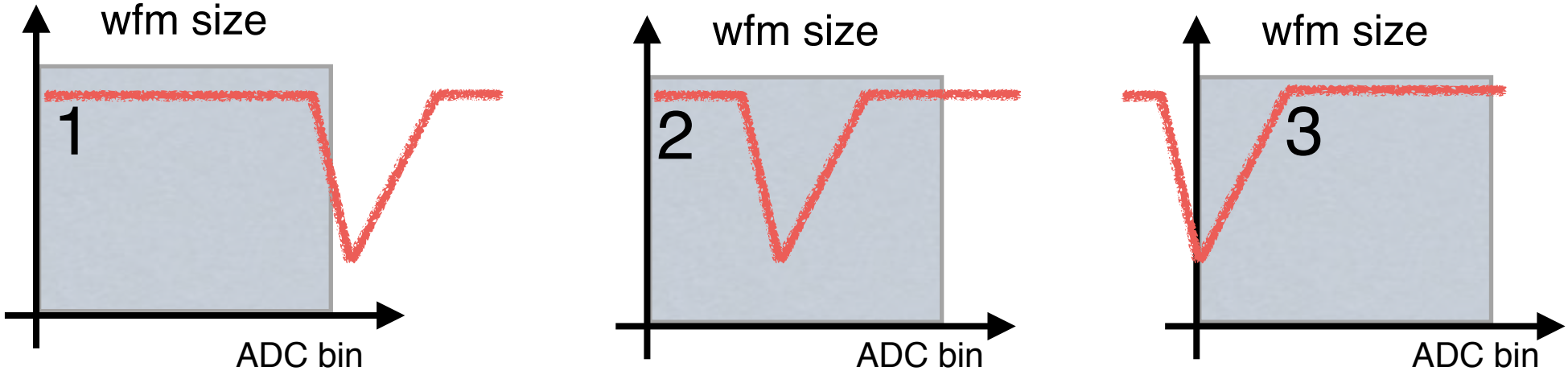
Trigger too late…



Rising edge lost

Need of a faster trigger or a delayed signal

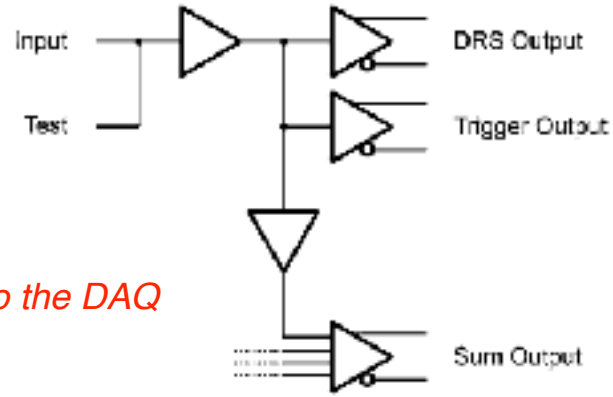
Waveform digitiser and trigger latency



- The DRS is continuously recording detector signals
  - *(1) a pulse is generated by the sensor and received by the front-end*
  - *(2) the pulse is well centred in the recording memory, the ADC bins in from of pulse are needed for waveform analysis!*
    - it is time to fire the trigger (if "good" event)
  - *(3) the pulse is being overwritten in the memory*
    - the trigger would be to late, the event could not be reconstructed
- The trigger latency is defined by ADC memory size
  - *is it possible to extend the memory? NO… it is defined by chip design*
  - *is it possible to delay signals? If possible NO… may spoil time resolution*

Trigger technology choice

- The trigger will be a separate system
- Latency requirements
  - *DRS with 1.6 GSPS (600ns waveform width)*
  - *~50 ns are required before the pulse in DRS waveforms for offline processing*
  - *~50 ns is a conservative estimate for the trigger signal distribution from the trigger to the DAQ*
  - *the decision must be taken as fast as 500 ns after the event occurrence*
    - at trigger level only fast detectors are used: LXe calorimeter and Timing Counter (plastic scintillator detector)



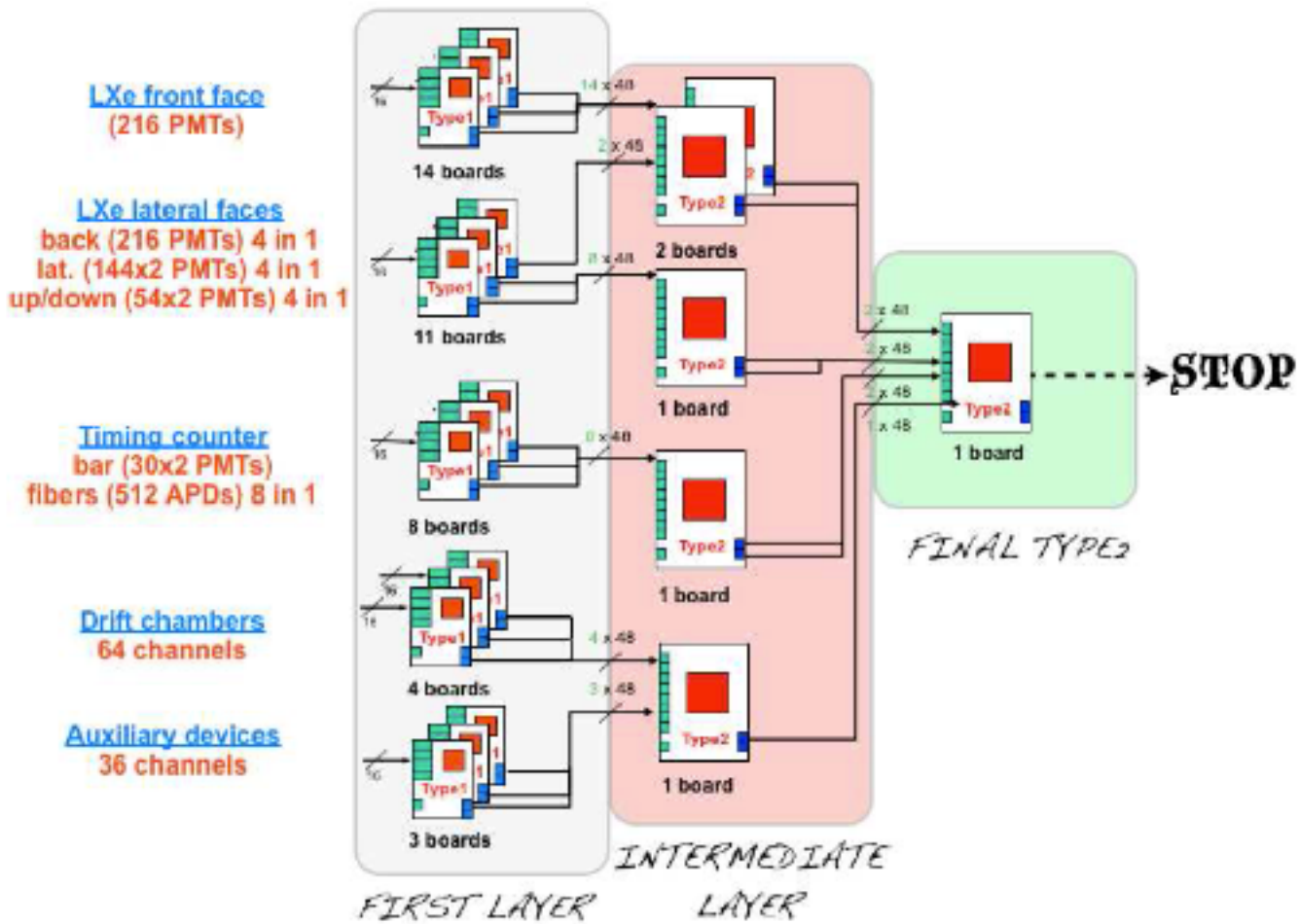
- Flexibility requirements
  - *the system have to cope with any possible experimental needs*
    - …and there are really a lot of!

- FPGA based trigger
  - *Xilinx Virtex-II pro FPGA (bought in 2004)*
    - algorithm execution frequency to be 100 MHz
  - *VME boards*
    - used for board configuration and monitoring @trigger level
  - *Multi layer system*
    - transmission with LVDS serialiser-deserialiser, 4.8Gbit/s per connection

Introduction to FPGAs
Hannes Sakulin

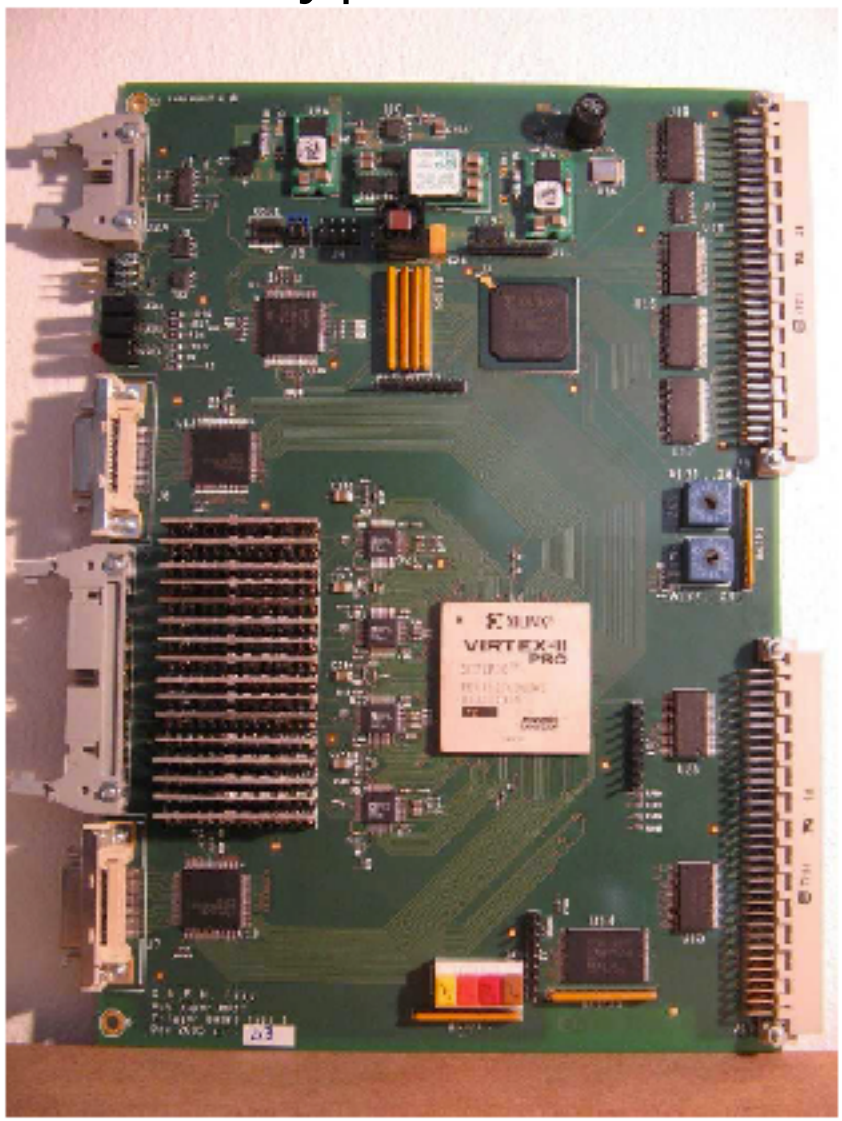
Overview of the trigger system

- Three layers system
  - *Type1 board*
    - 6U VME board
    - receiving detector data
    - equipped with FADC 100MHz - 10 bit
  - *Type2 board*
    - data processing and transmission to next level
      - *the master performs last algorithm steps and fires the trigger signal*

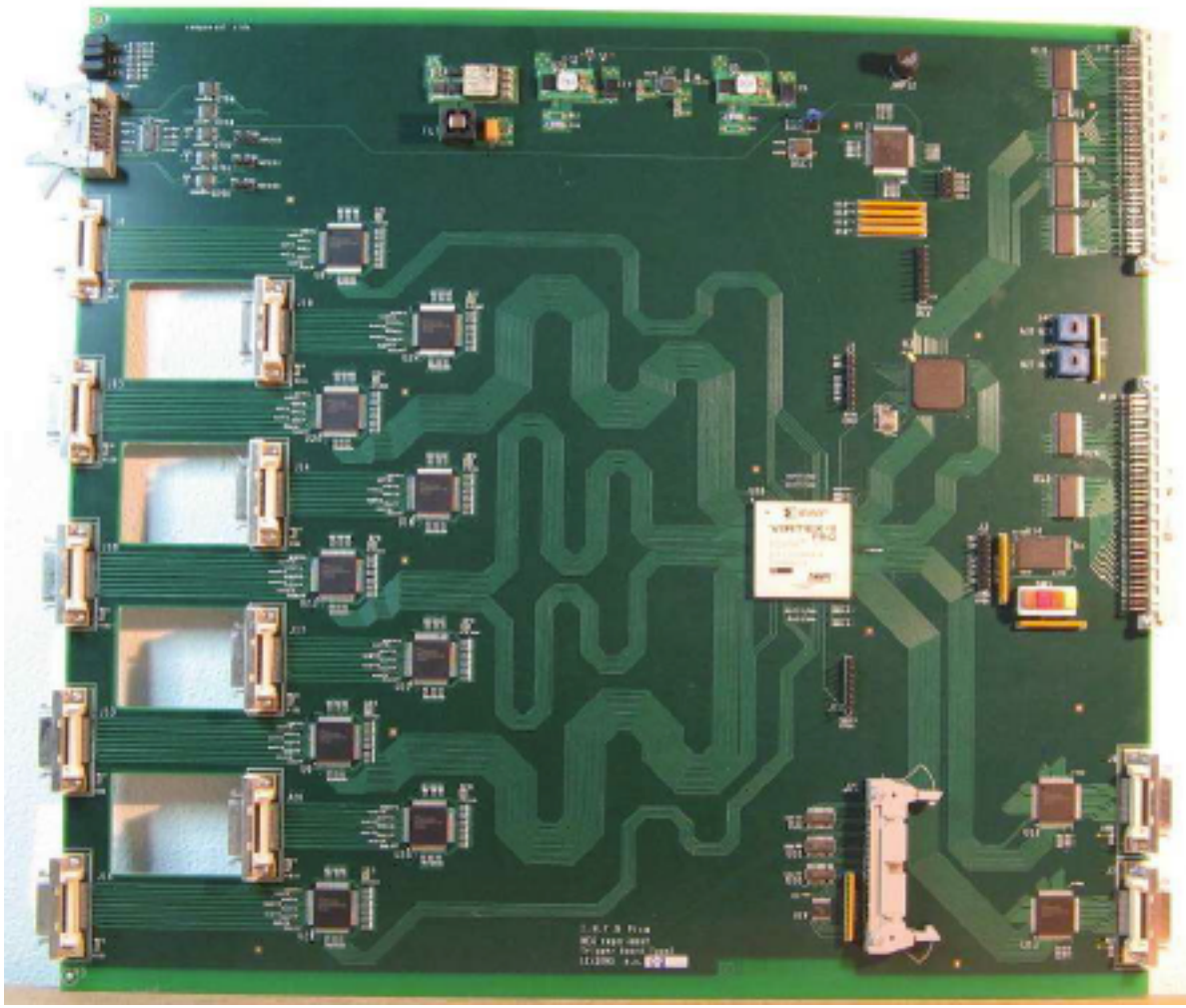


Some pictures… trigger boards

Type1-6U



Type2-9U

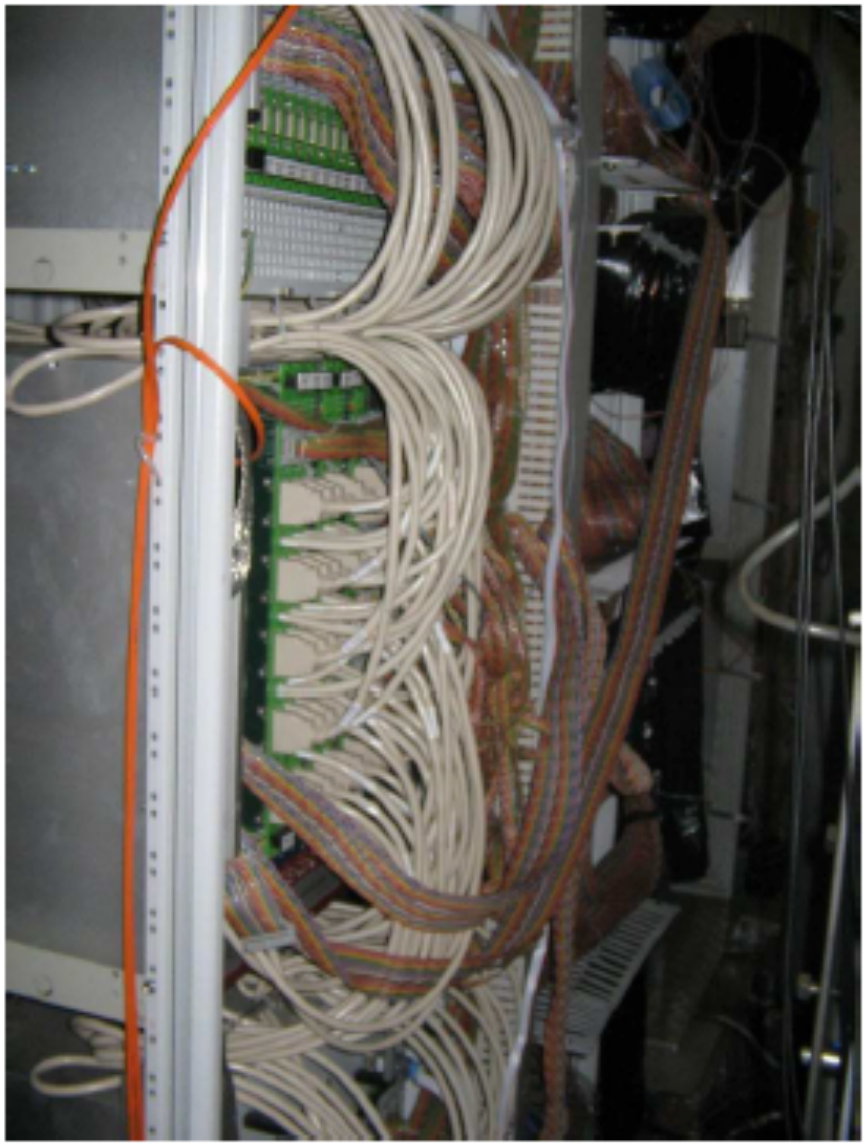


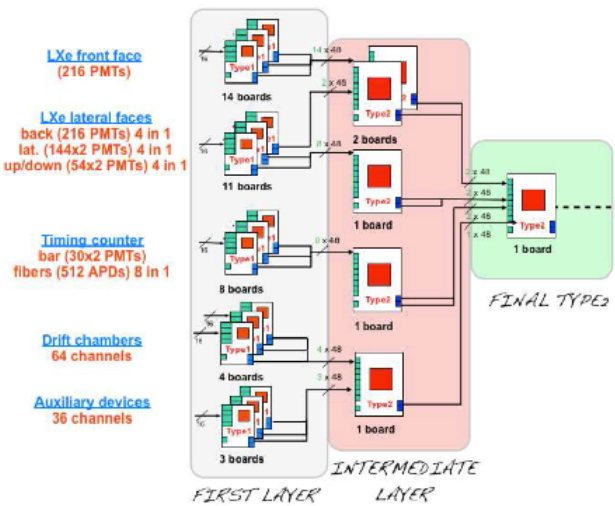
The trigger system at PSI

Type1 6U crate

Type2 9U crate

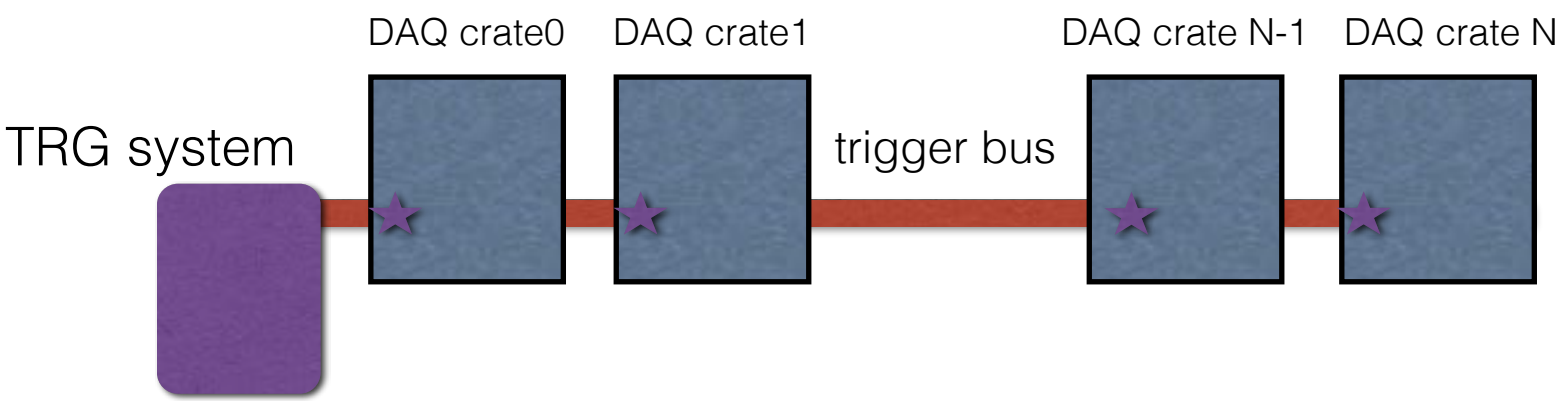
Type1 6U crate





Trigger bus and event building

- The raw data are scattered over several VME crate
  - *each crate is read-out after any trigger by dedicated CPU that create a fragment*
  - *all the fragments sent to the main CPU*
    - what happens if at a certain point a fragment is loss??
      - *fragment belonging different events are mixed*
      - *data useless!!*

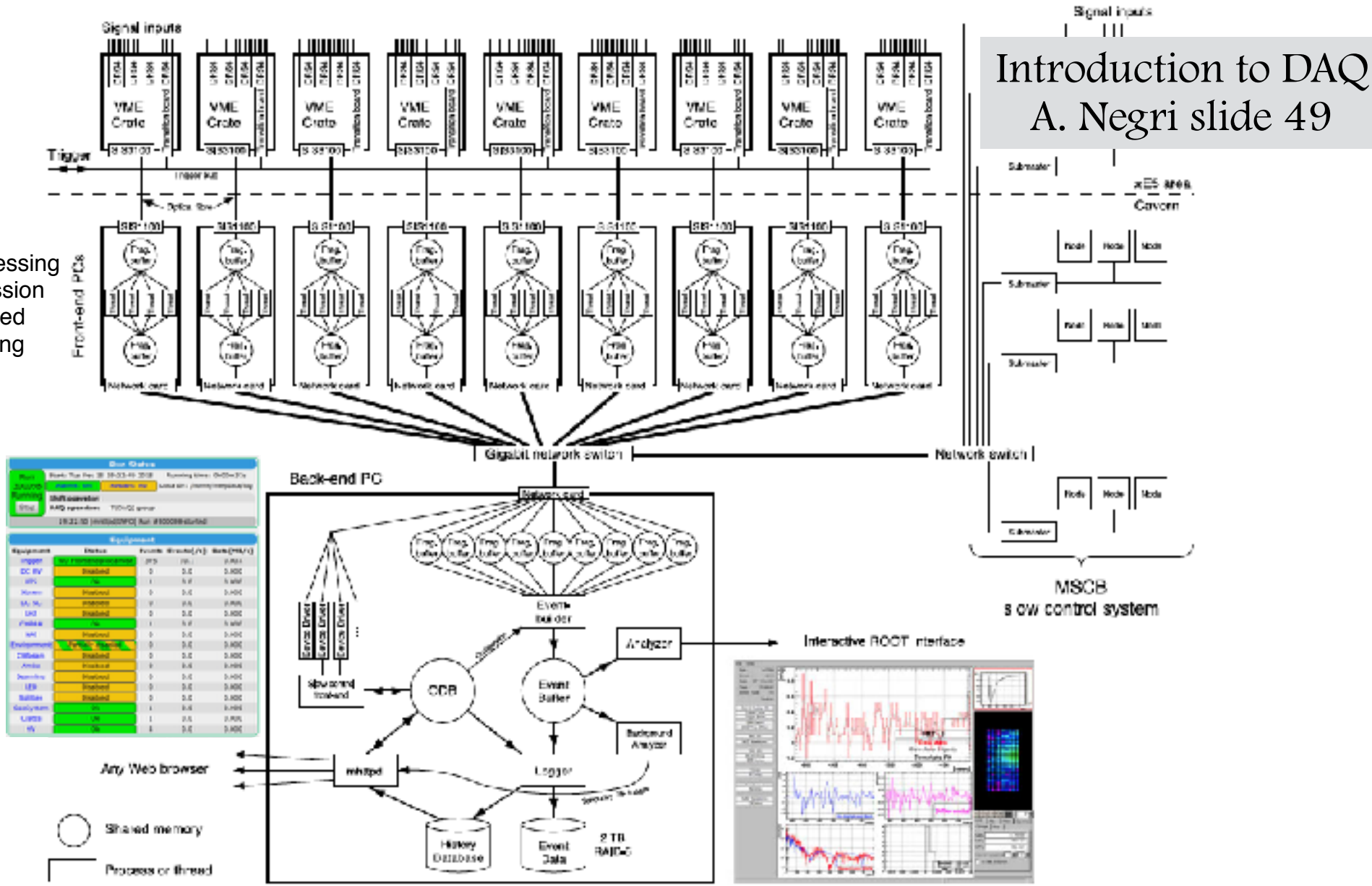


- The solution is the trigger bus: from the trigger to the DAQ crates
  - *event number + event code (trigger type)*
  - *connected via the VME transition boards on the backplane*

Read out architecture - MIDAS

Introduction to DAQ
A. Negri slide 49

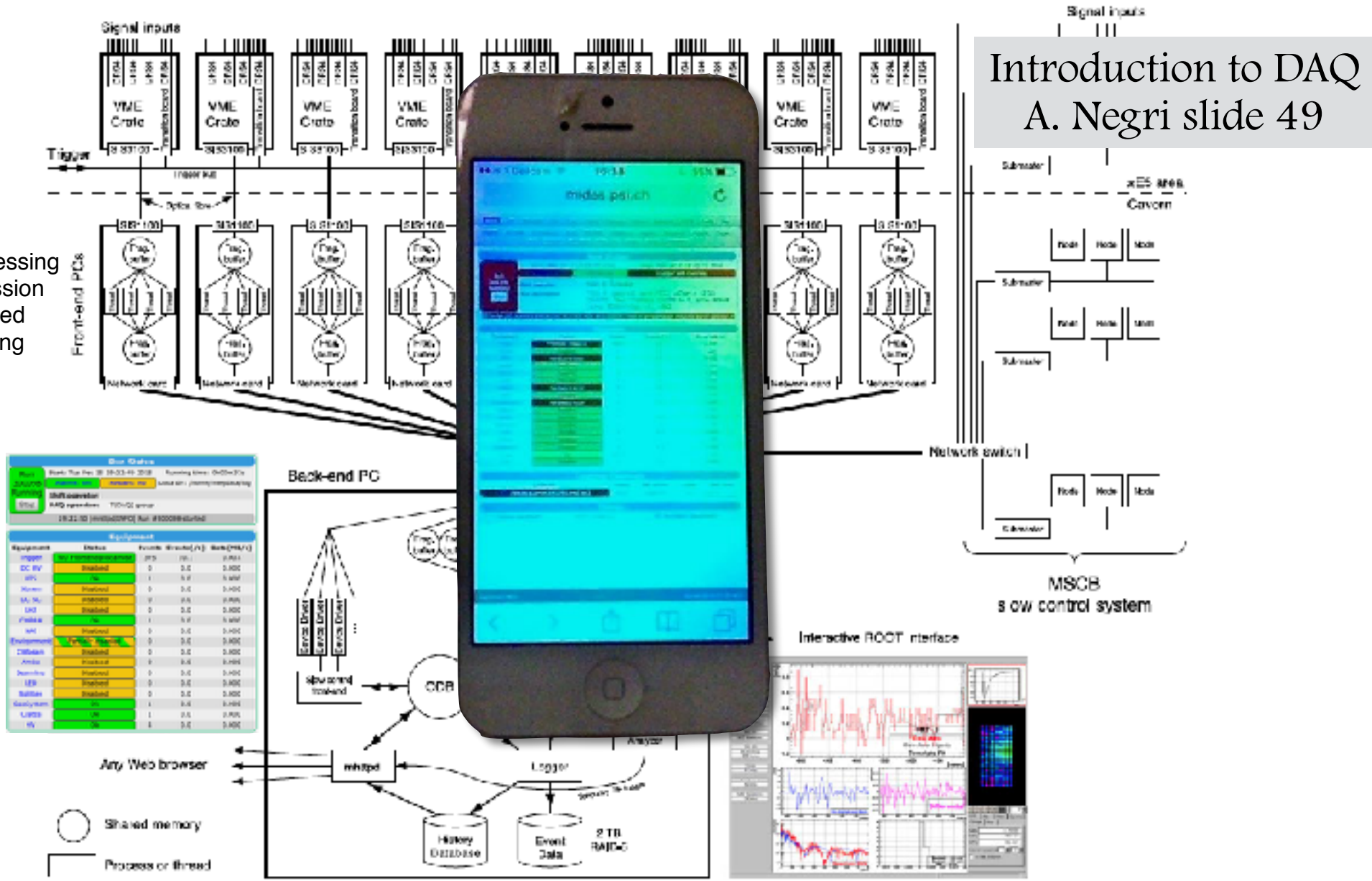
Data pre-processing zero-suppression with threaded programming



Read out architecture - MIDAS

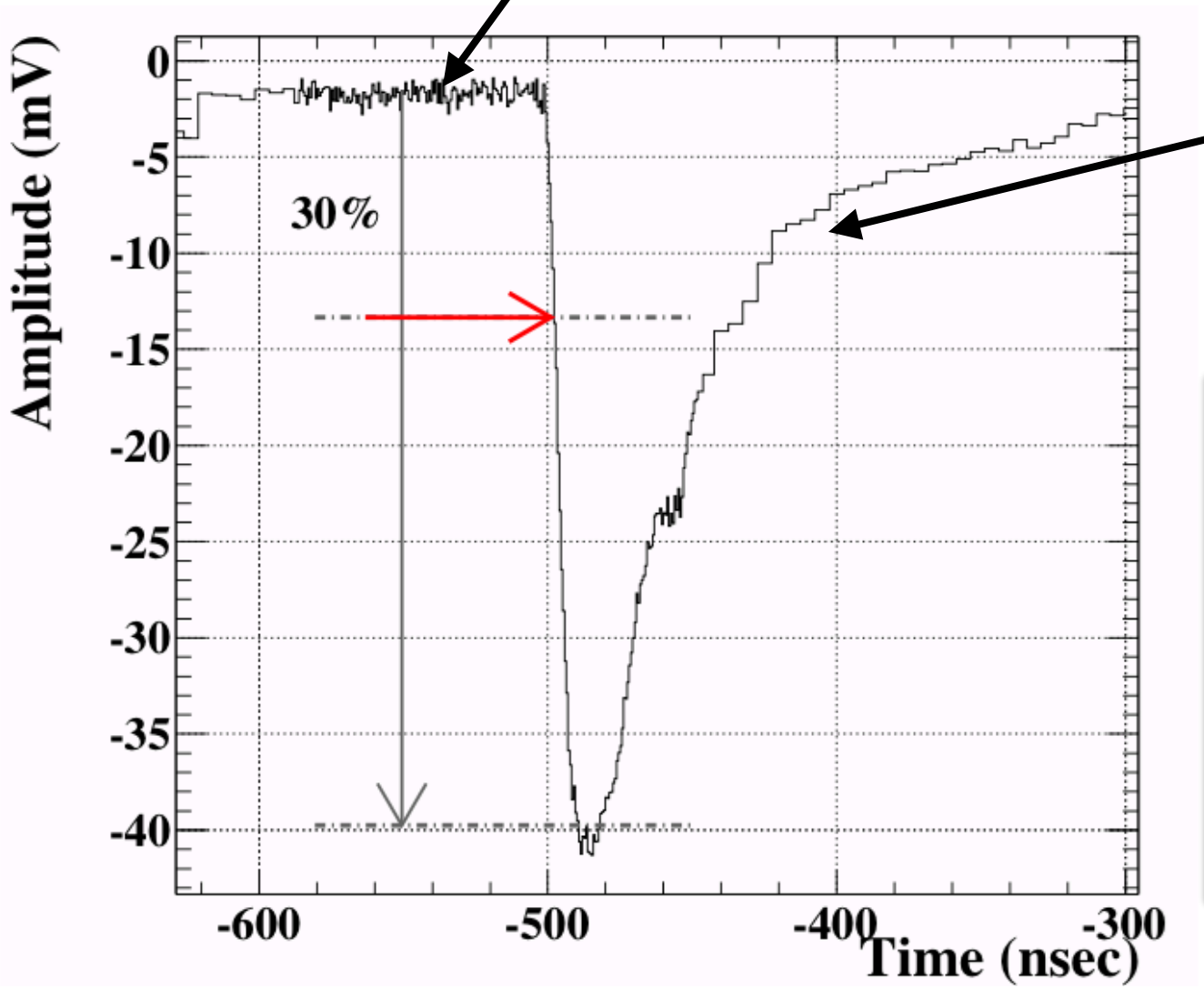
Introduction to DAQ
A. Negri slide 49

Data pre-processing zero-suppression with threaded programming



A zero-suppression example

1.6 GSPS



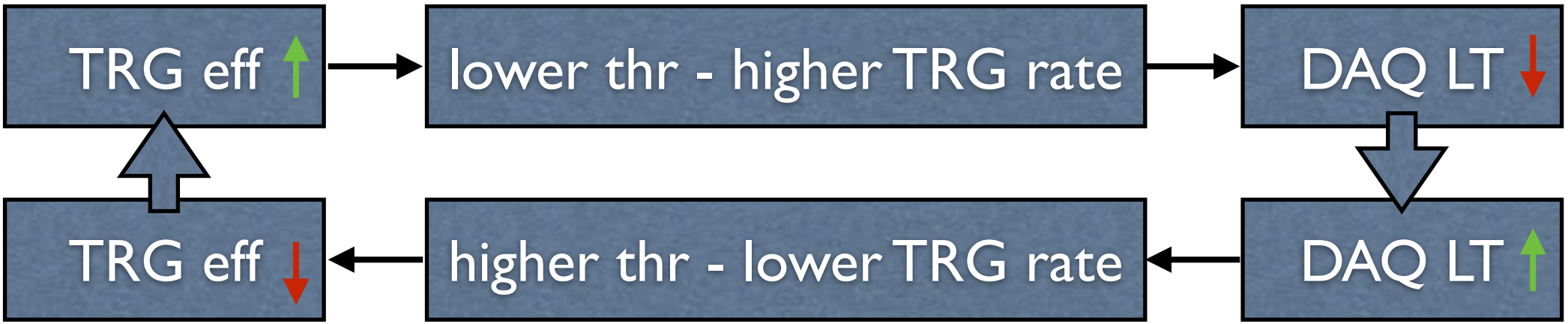
re-binned
by factor 10

Almost a factor 2 gain in data size BUT no ~information lost

No ultimate re-binning (QDC-like) to be sensitive to pile-up

Some consideration about DAQ efficiency

- DAQ efficiency is defined as
  - *DAQLiveTime x Trigger Efficiency*
- This quantity has to be as large as possible in any TDAQ system



- Compromise…
  - *milestones! let's start from dead time…*

Read-out dead time

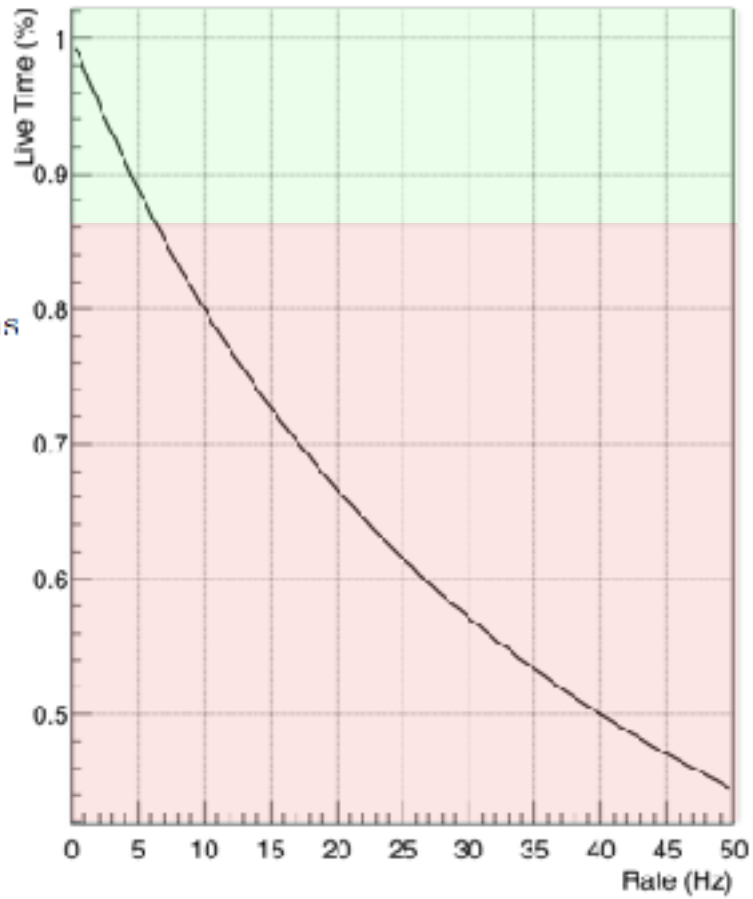
- As a first attempt let's try with a single buffer read-out
  - *the system is in dead time during the read-out of a crate through VME*
  - *the dead time is dictated by the heaviest crate*

$$t_d = \underset{\#\ \text{mezzanines/crate}}{40} \times \Big(\underset{//\ \text{DMA setup time}}{0.125\,\text{ms}} \times 40960\,\text{B}/83000\,\text{MB/ms} \Big) = 24.7\,\text{ms}$$
$$\#\ \text{data transfer time}$$

- the associated DAQ live time is given by the probability to have 0 event during the read-out (trigger rate = 7 Hz, Poisson event distribution)

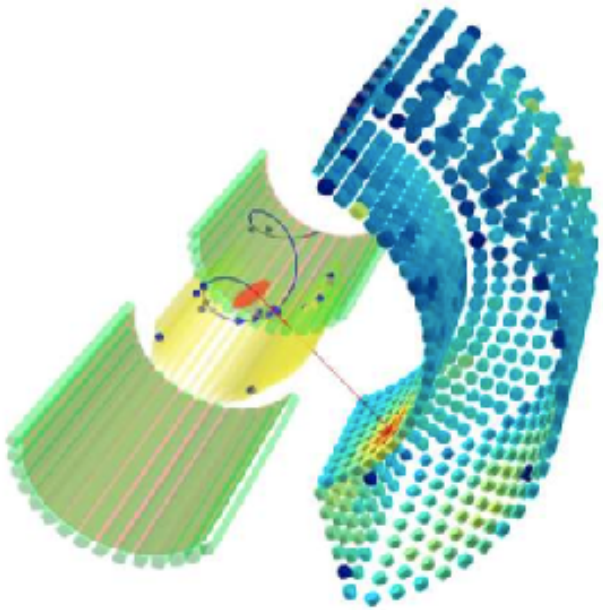
$$LT = \frac{1}{1 + t_d \cdot R_{trg}}$$

- Requirement for the trigger rate
  - *which trigger algorithms can we use?*
  - *what about the rejection power?*



Considerations about trigger algorithms

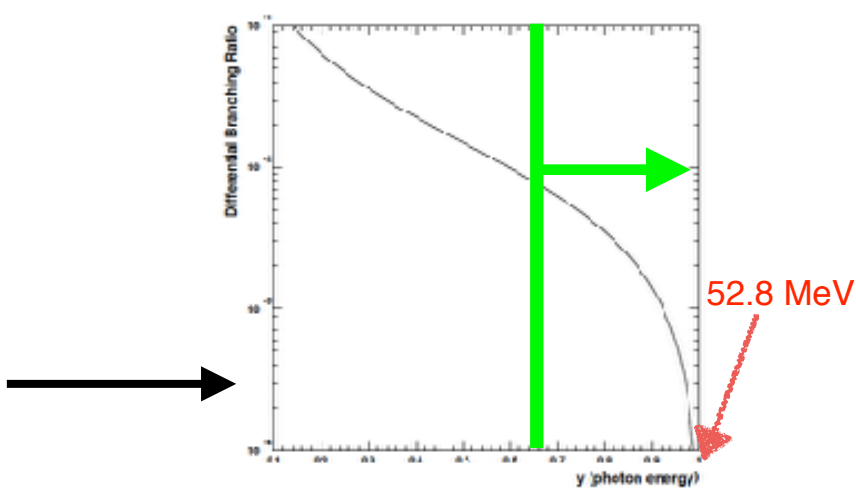
- Goal: DAQ rate of about 7 Hz
- Use of prompt response detectors
  - *LXe calorimeter (photon energy, time and direction)*
  - *Timing Counter (positron time and hit position)*
- Synchronous processing with FPGA
  - *algorithms to use 1 CLK cycle per operation (if possible)*
    - registers, adders, subtracters, comparators for simple operations
    - Look Up Tables for more complex operations, for example products



This choice is driven by physics and your detector!

- Online observables
  - *photon energy*
  - *photon-positron timing*
  - *photon-positron space correlation*

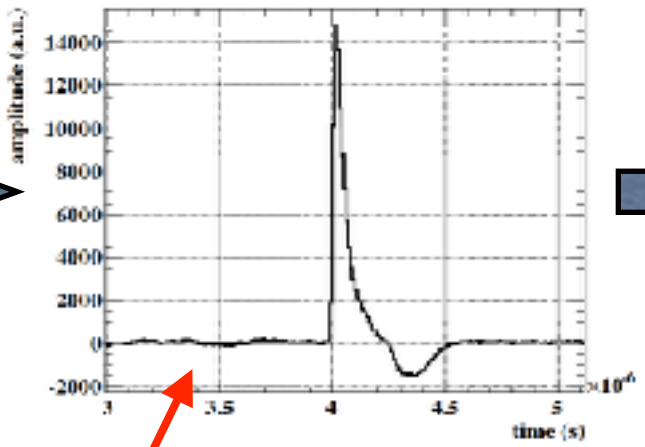
- The discrimination on photon energy has the largest background rejection power
  - *take care about it!*

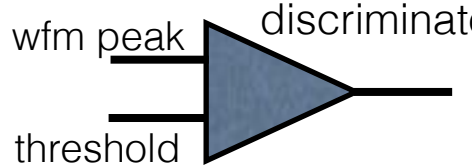


Online energy reconstruction

- Synchronous sum of the LXe waveform samples
  - *the peak of the obtained waveform is the online energy estimate*

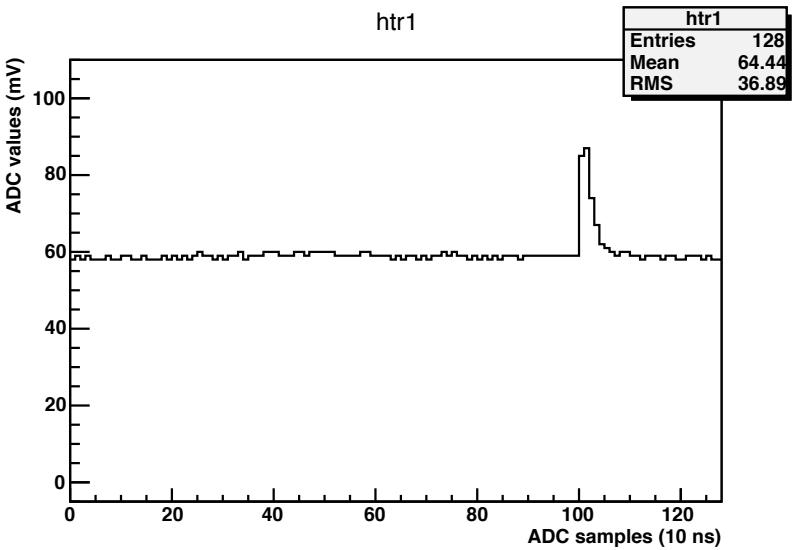
$$A_{tot}(t) = \sum_{i=0}^{N_{CHA}} f_i A_i(t)$$

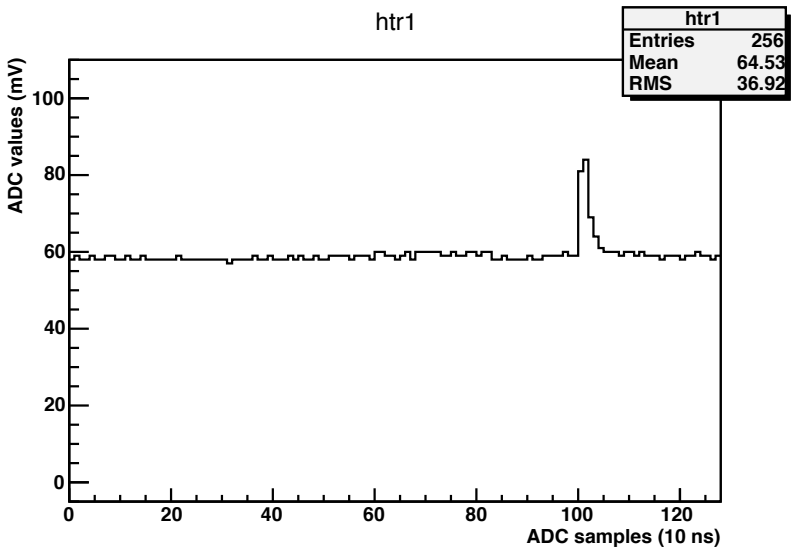


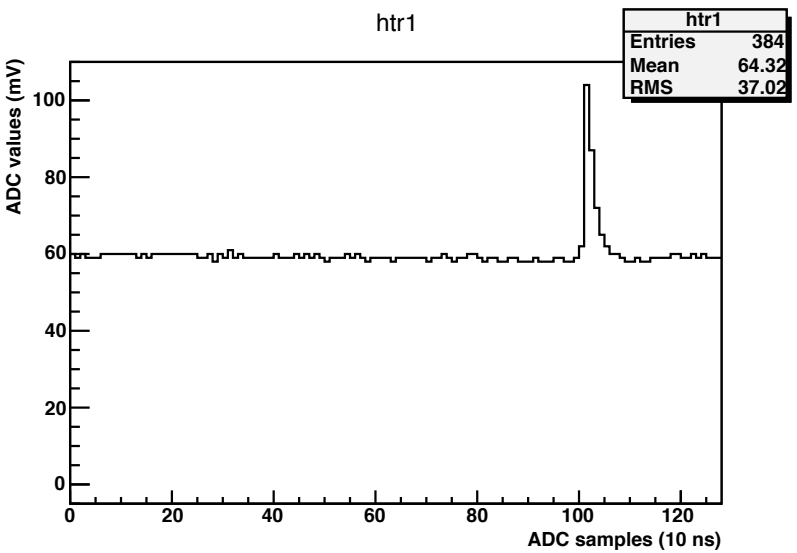


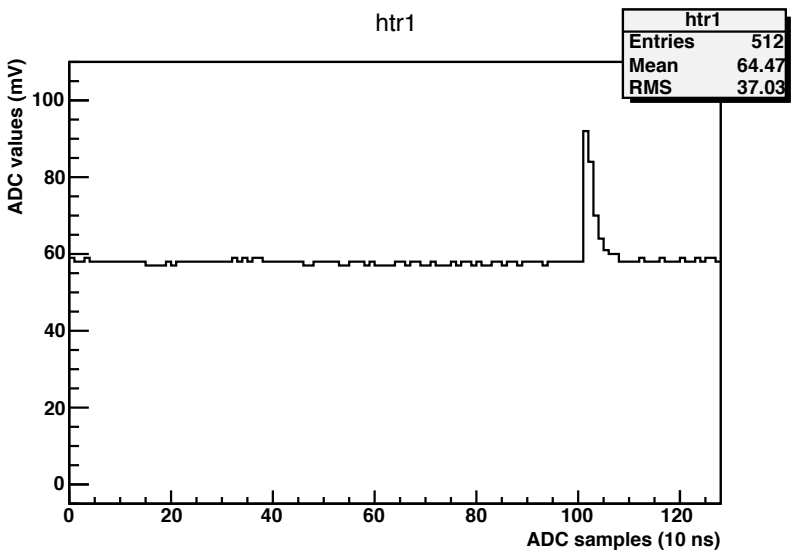
- Algorithm highlights
  - *online pedestal subtraction*
    - the baseline value may differ from channel to channel (remember QDC in lab4)
      - *refer the waveform to a common value (0 ADC counts) before the sum stage*
  - *channel calibration with Look Up Table*
    - for example in case of PhotoMulTipliers gain and Quantum Efficiency calibration

Raw waveforms from ADC

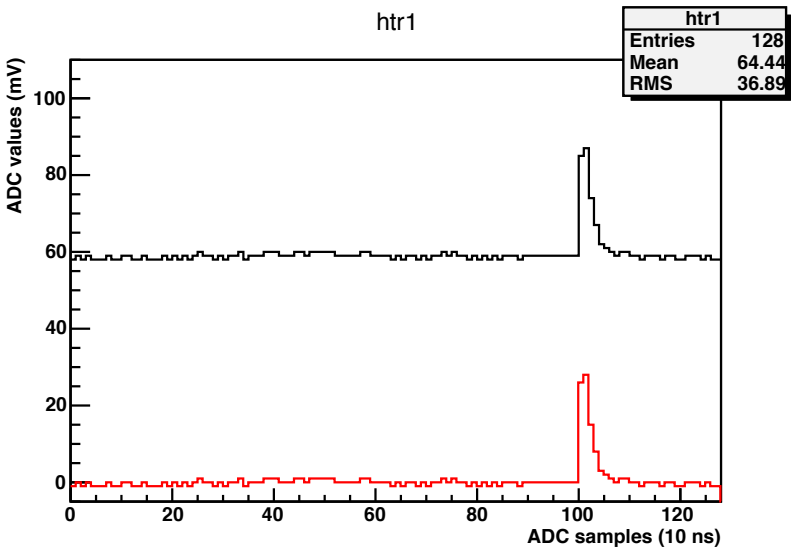


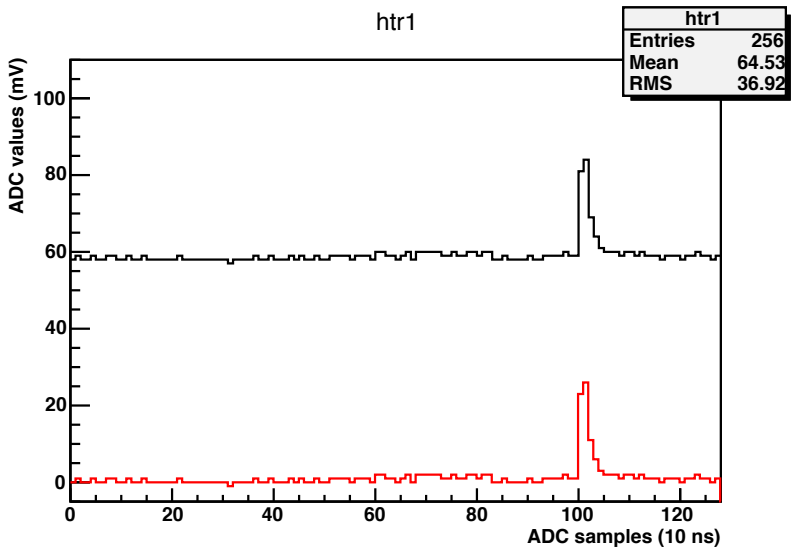


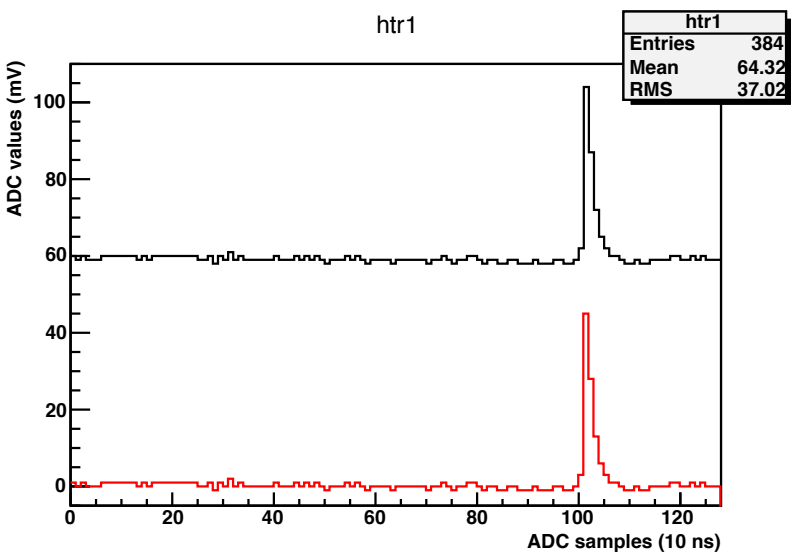


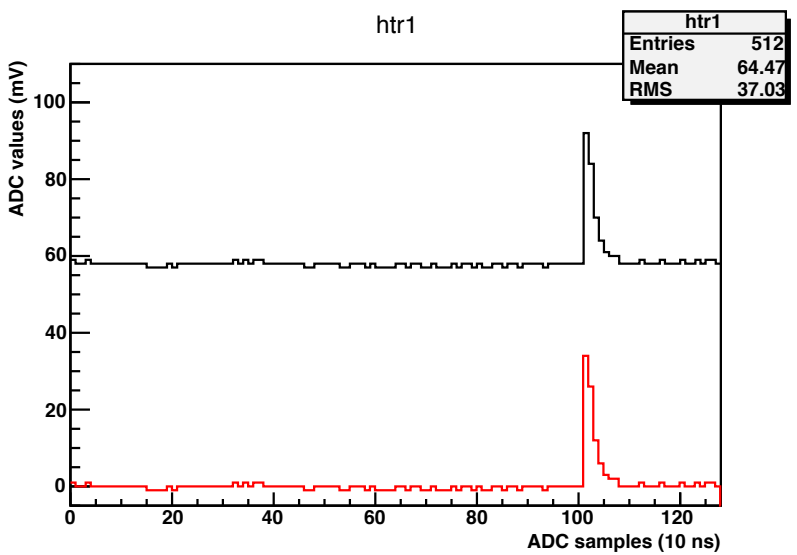


…online pedestal subtraction…



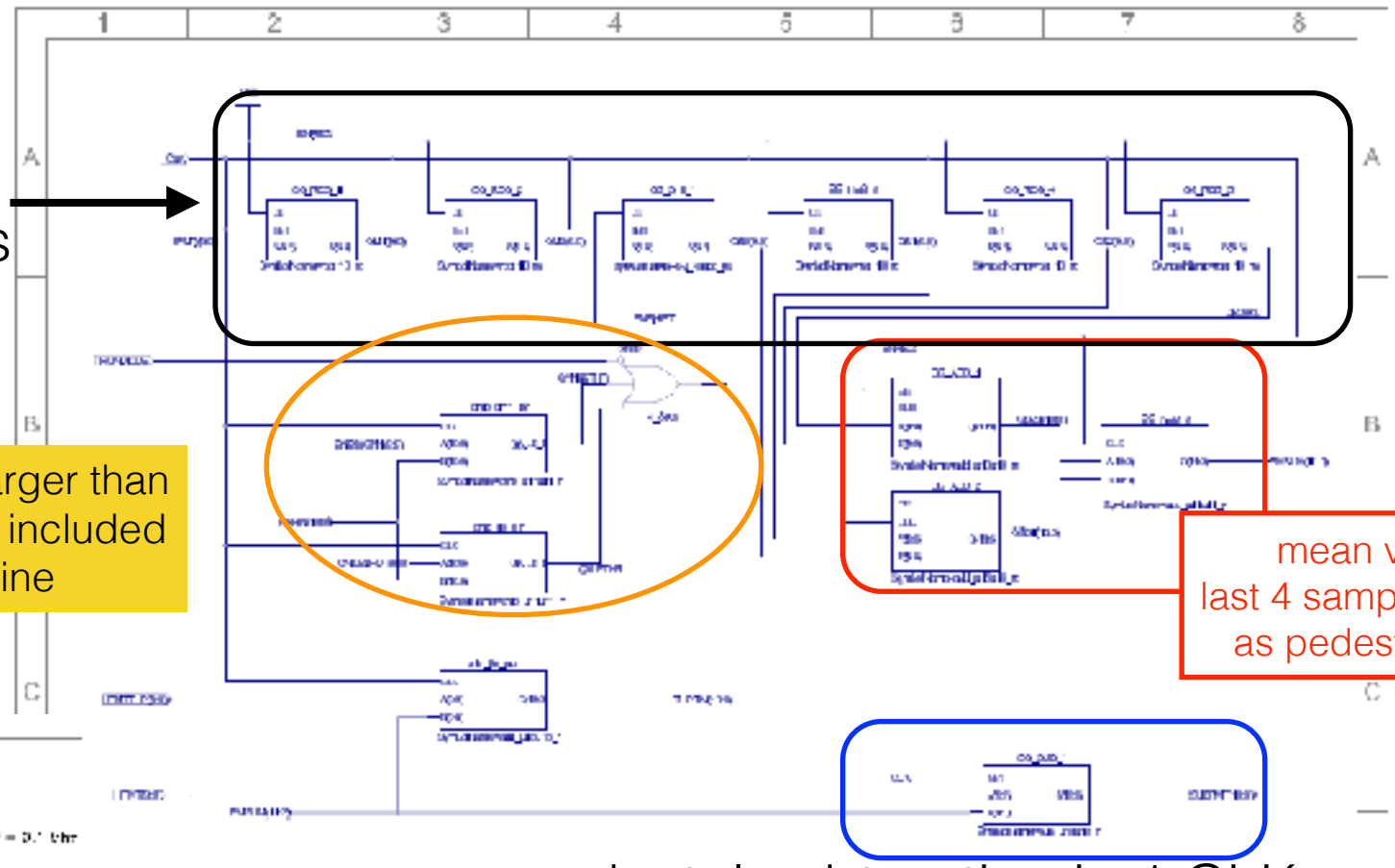






Firmware for online pedestal subtraction

pipeline with shift registers

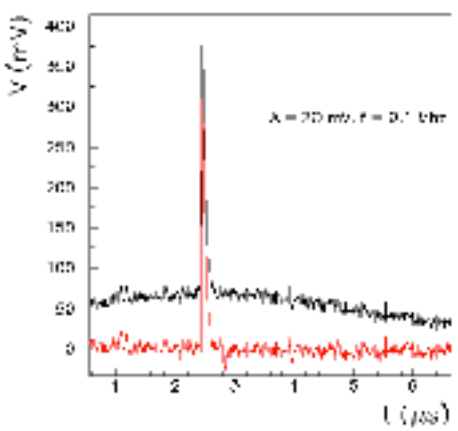


Algorithms with schematic language

VME protocol + some algorithm with verilog

If the sample is larger than threshold it is not included in the pipeline

mean value from last 4 samples in pipeline as pedestal estimator

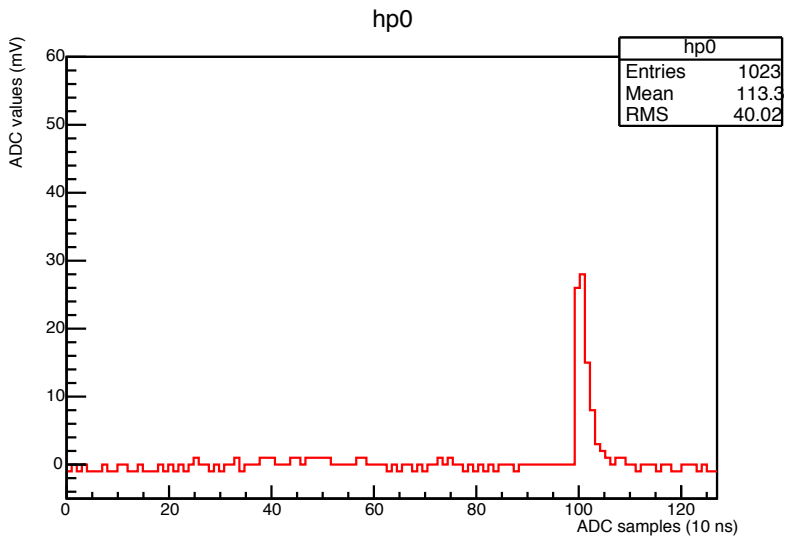


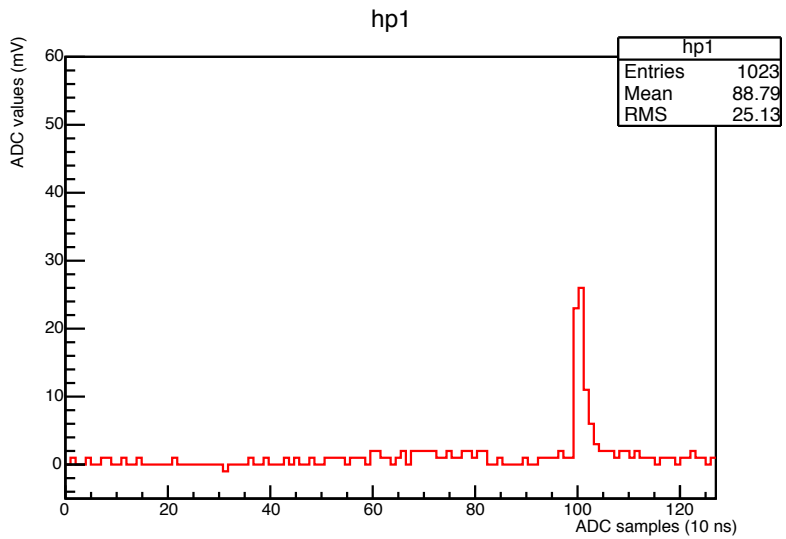
pedestal subtraction in 1 CLK cycle!!

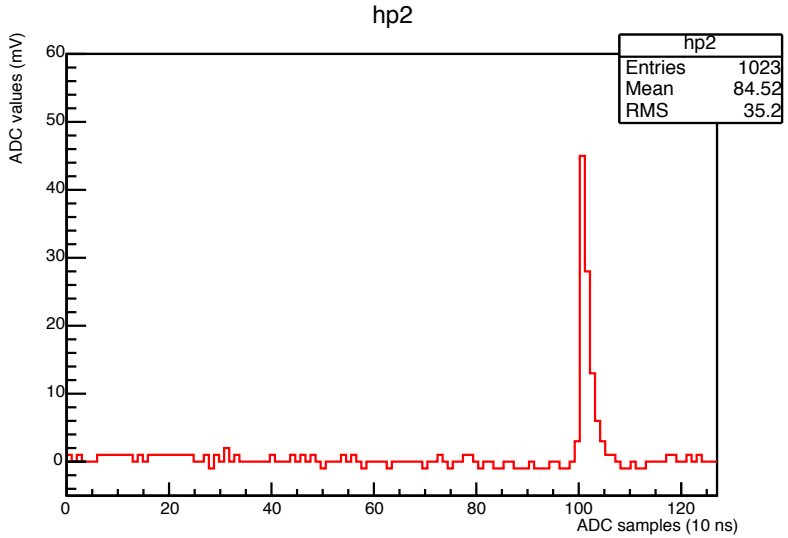
10 ns

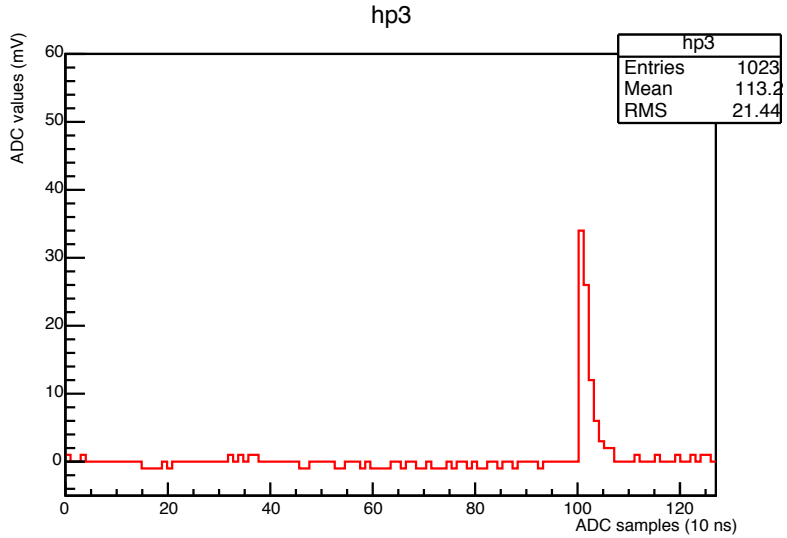
Advanced FPGA programming
Manoel Barros Marin

…again pedestal subtracted wfms..

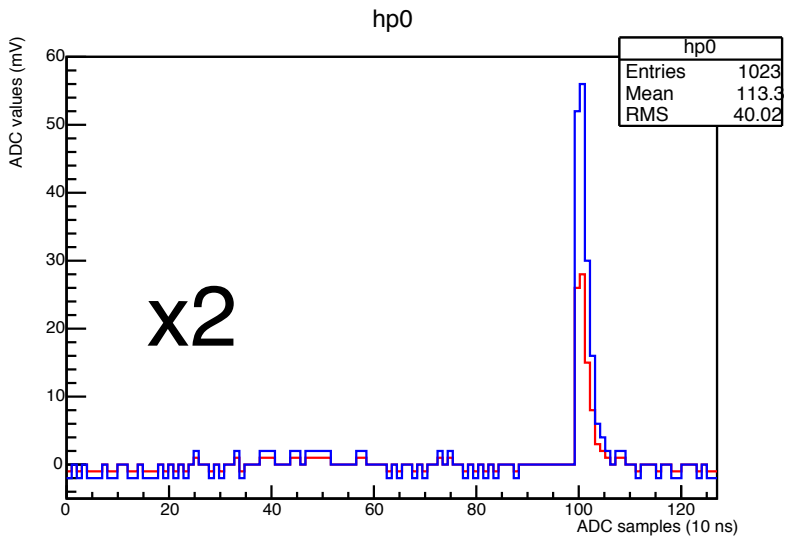


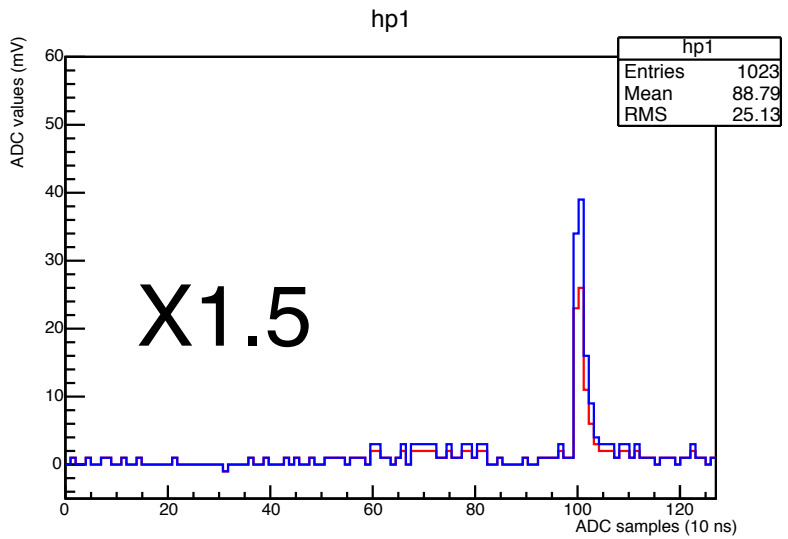


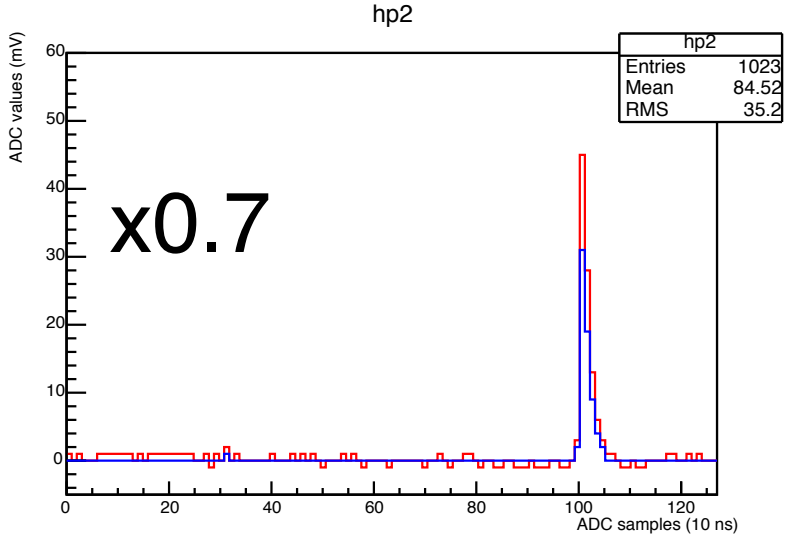


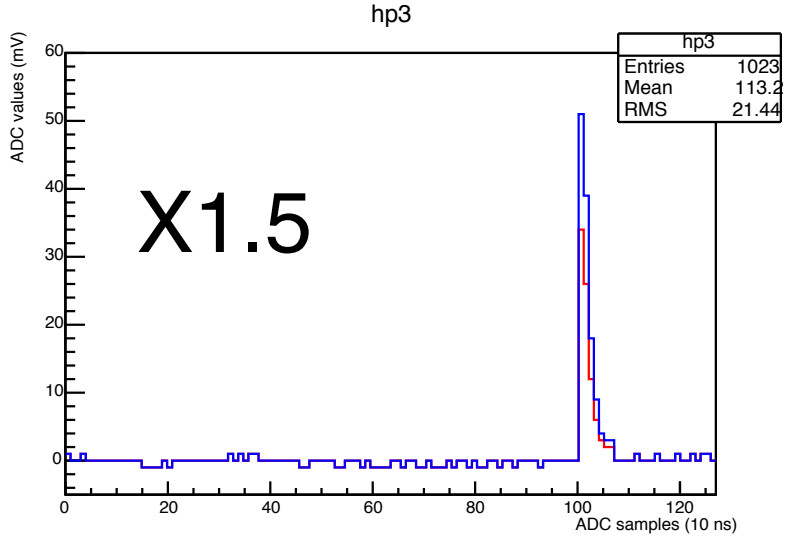


…channel calibration…

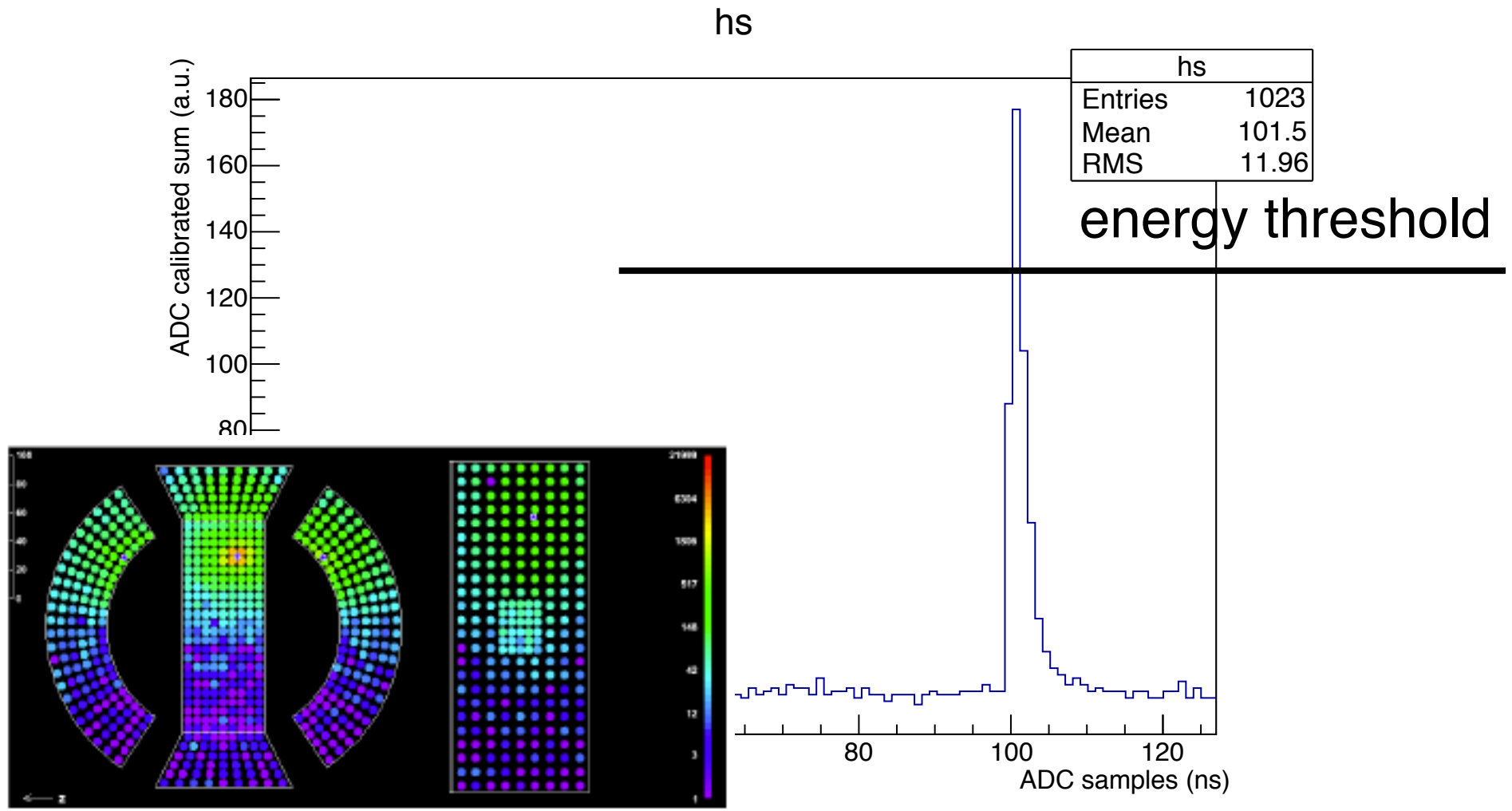






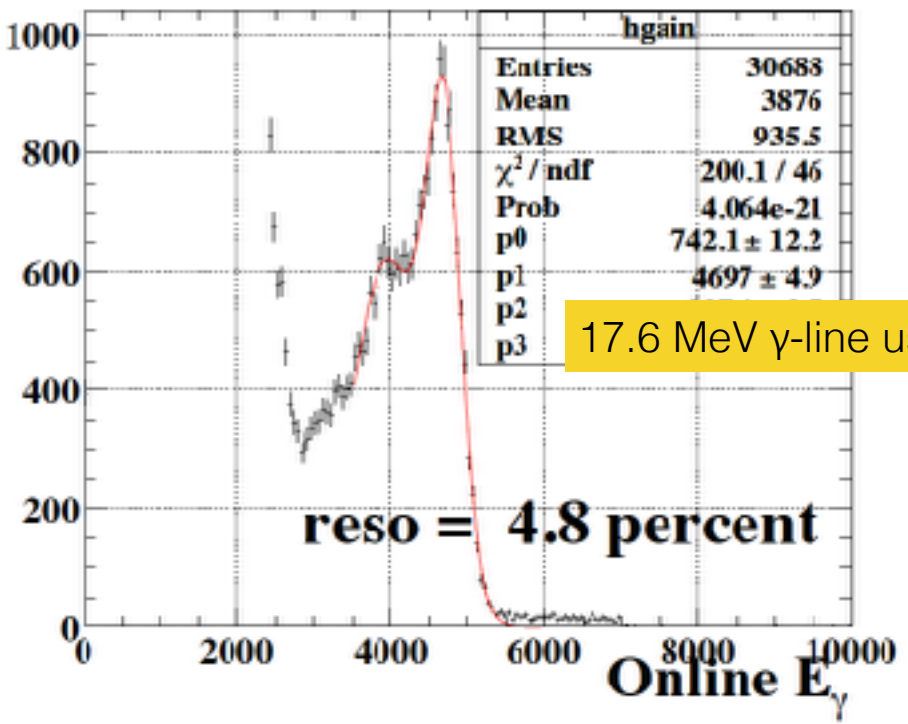


…and finally coherent sum

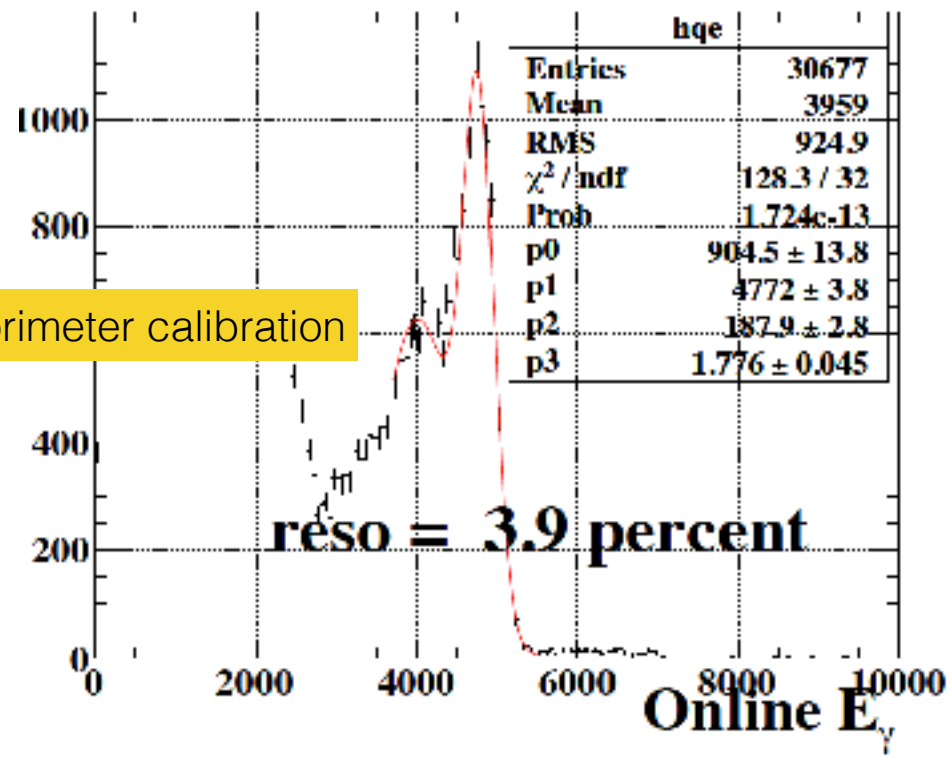


Reconstruction calibration!

- By applying a refined calibration reconstruction improves!
  - *better reconstruction -> higher threshold with same efficiency on signal -> lower trigger rate -> higher DAQ Live Time with the same trigger efficiency*
    - improvement of the DAQ efficiency



17.6 MeV γ-line used for calorimeter calibration

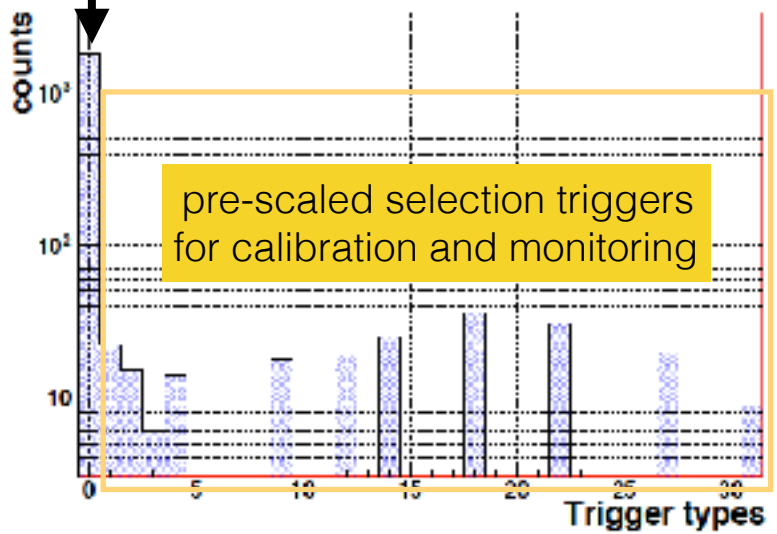


- It is important to get the best from the system with the available techniques!

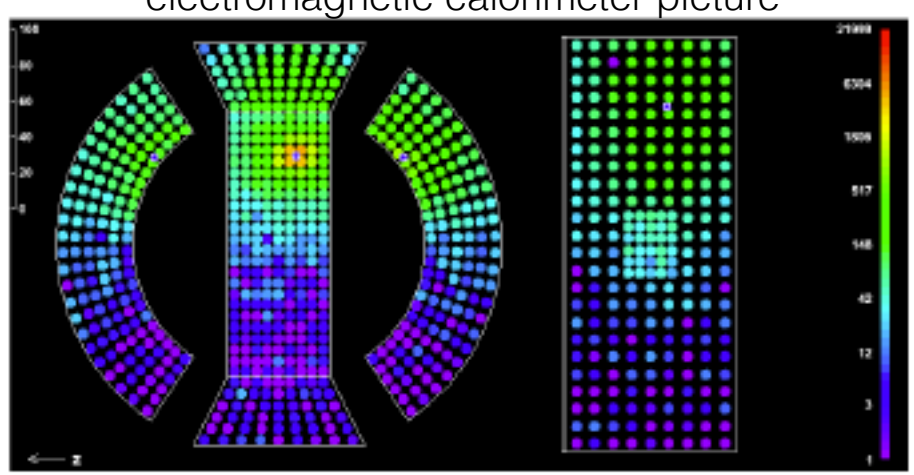
Run configuration and monitoring

- The trigger system provides the DAQ with 32 different selection algorithms
  - *the trigger type are order by priority*
  - *pre-scaling factors*
    - programmable fraction of minimum bias selection in the data stream
      - *trigger efficiency studies*
      - *detector calibration and monitoring*
      - *physics analysis, normalisation evaluation*
- This flexibility is crucial to take under control the detector
  - *the run configuration is store in a database*
    - give access to the shift crew
    - default run configuration available
    - metadata for analysis (as discussed in lab 4)
- GUI to check data quality online
  - *events "pictures"*
    - not so precise but sensitive to major problems
- Automatic analysis right after the acquisition to check further the data quality
  - *histograms for detector debugging*
    - dead channels, electronic noise…

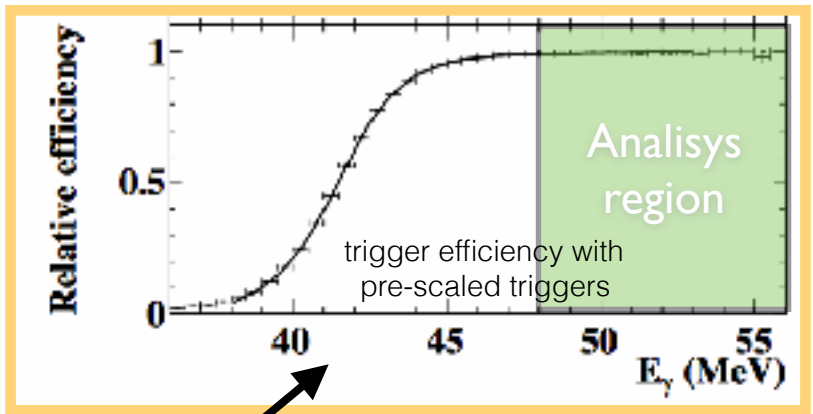
main selection
95% of triggers



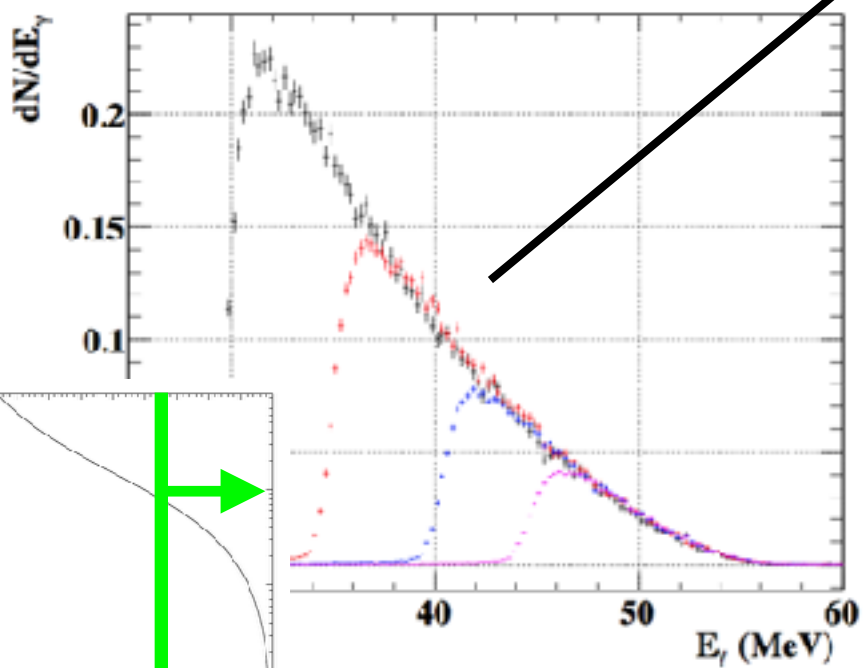
electromagnetic calorimeter picture

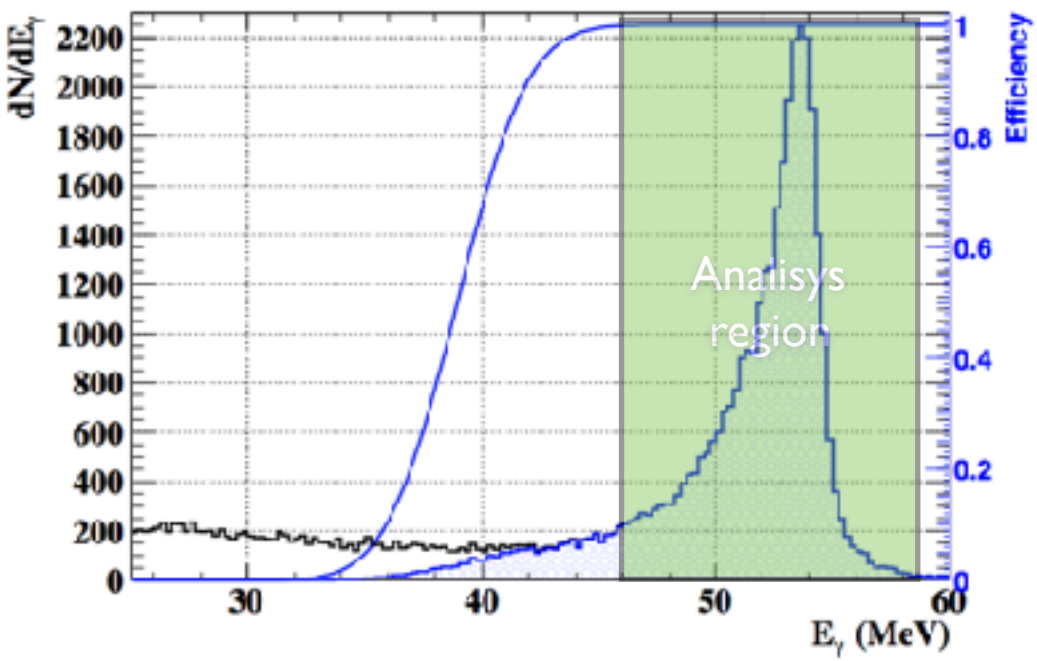


Trigger efficiency



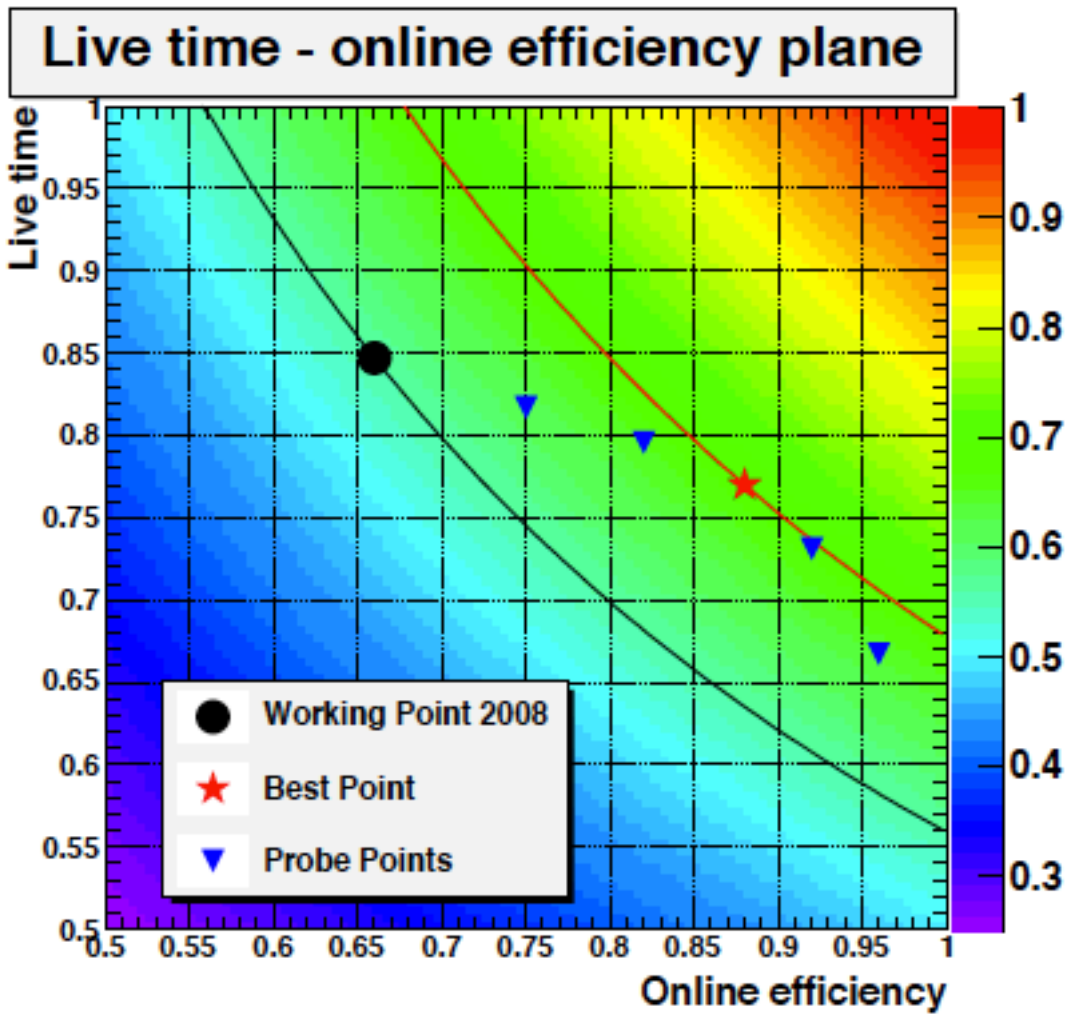
Error function from the ratio of spectra acquired with different energy thresholds





DAQ efficiency: measurement and optimisation

- First run in 2008: DAQ efficiency = 55%
  - *"preliminary" trigger system configuration*
    - first run with a complete detector
    - calibration not optimised yet
- The DAQ (trigger) efficiency measured at the end of data taking
  - *the selection has to be trained with data*
  - *the higher Live Time is NOT the experimental working point*
    - but the Live Time is measured online…
- How to find the best working point?
  - *trigger "simulation" with real data*
    - modify selection and predict the trigger rate and efficiency with no algorithm improvement
  - *improve the selection algorithms*
    - for example improve calibrations!
- By algorithm refinement we reached 75% DAQ efficiency



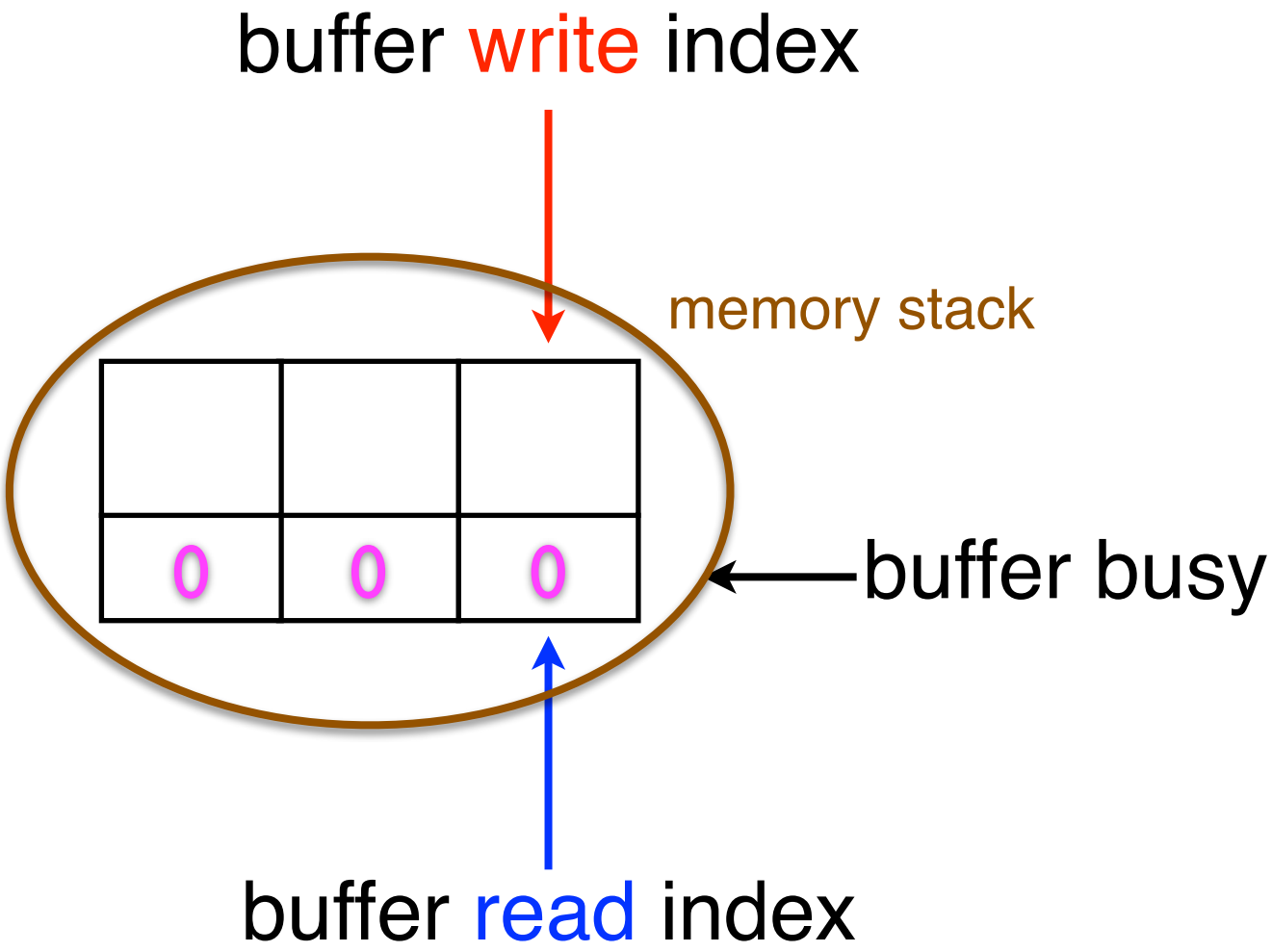
Is it possible to improve further?

- Many possibilities
  - *reduce dead time read out*
    - zero-suppression of FPGA to neglect the read-out of “empty” channels
      - *this is dangerous with an homogeneous calorimeter…*
    - Use 2eVME D64 read-out (160 MB/s instead of 80 MB/s)
      - *you have to foreseen it at design phase…*
        - unfortunately you usually miss something in your original design

Lab. 1 VME bus

  - *more complex selection algorithms*
    - current latency of the order of 500 ns, DRS running at 1.6GSPS
      - *a more complex algorithm (charge instead of pulse height) would lead to a larger latency…*
  - *Multiple-buffer read-out!*
    - data stored in circular memories on VME boards
      - *during a buffer read-out the next buffer is filled (if free…)*
      - *busy released immediately!*
        - this can be implanted in FPGA in any time… if you have enough resources!

Multiple buffer(0)



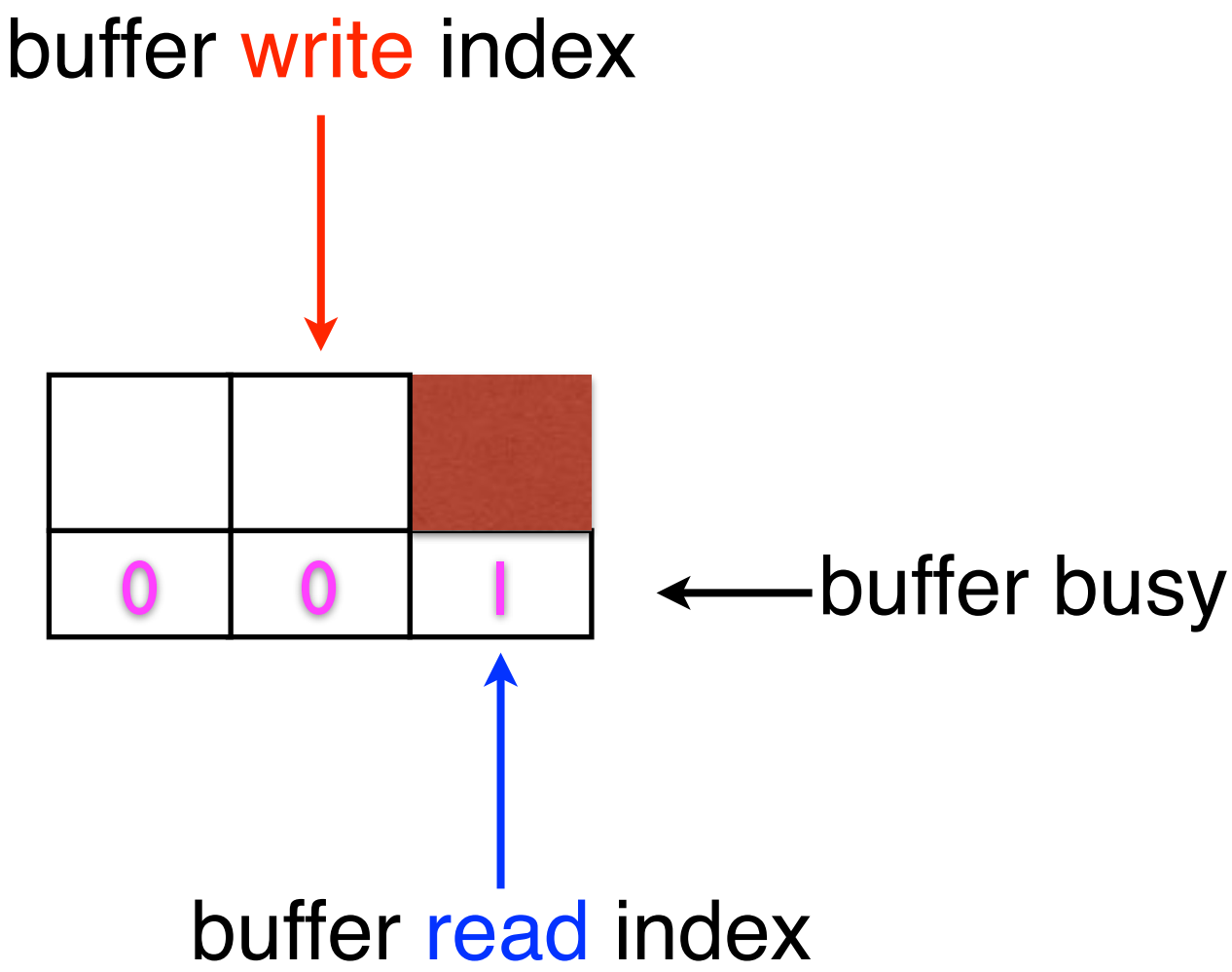
Introduction to DAQ
A. Negri slide 30

Situation at the begin of a run

all buffers free

all BUSY at 0

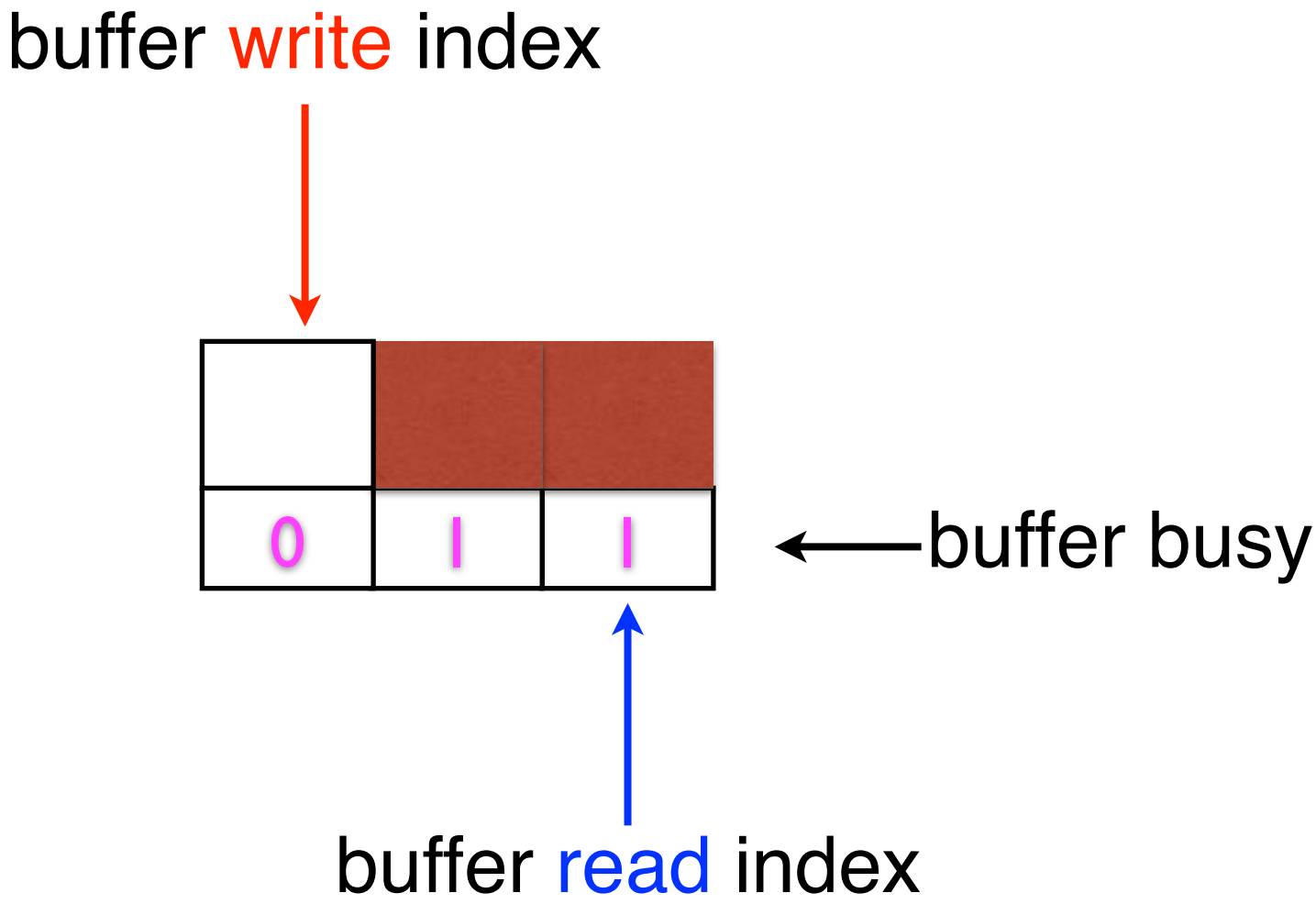
Multiple buffer(1)



Situation after the first event

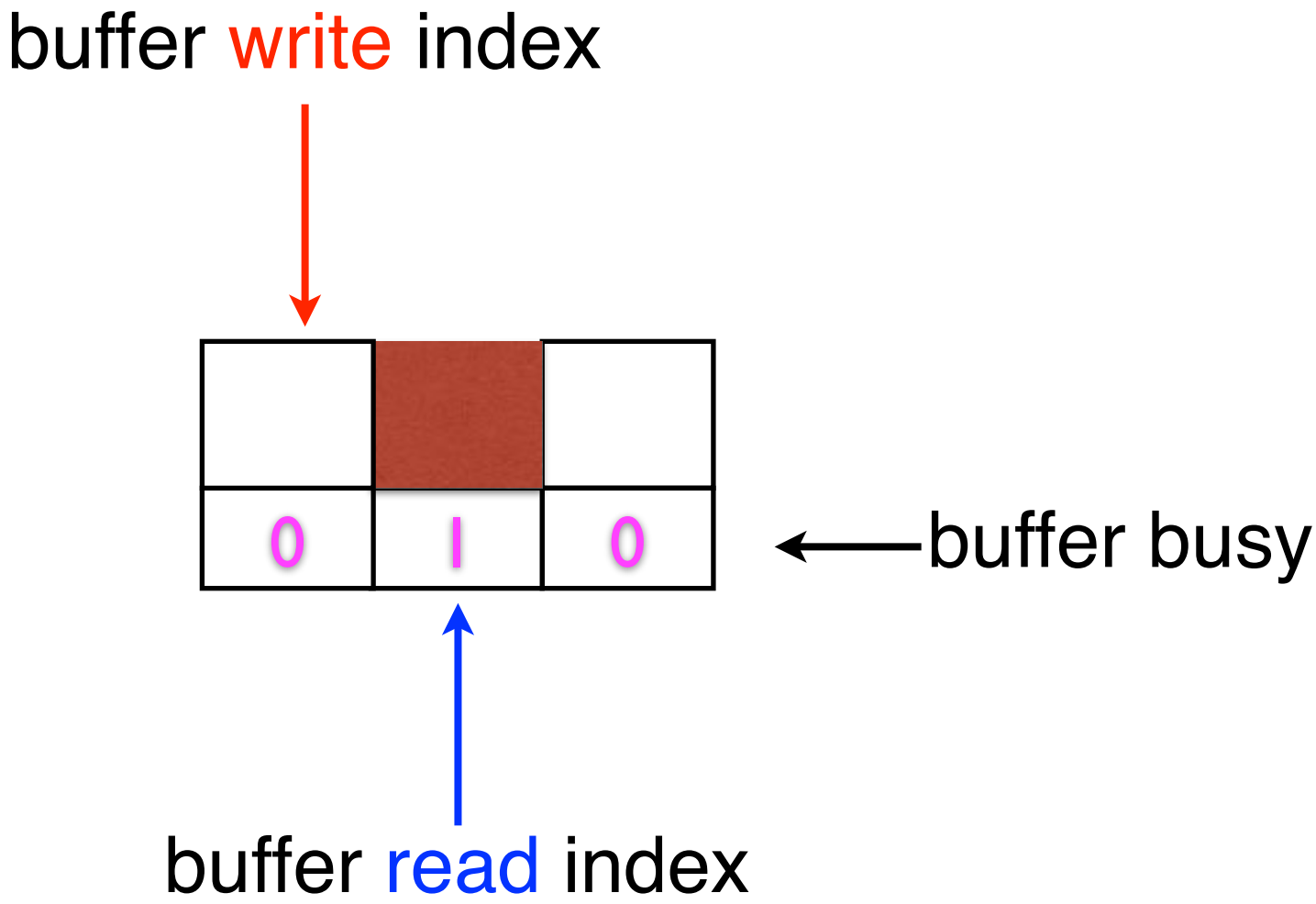
1 buffer full filling the next one

Multiple buffer(2)



Another event during the first buffer read-out

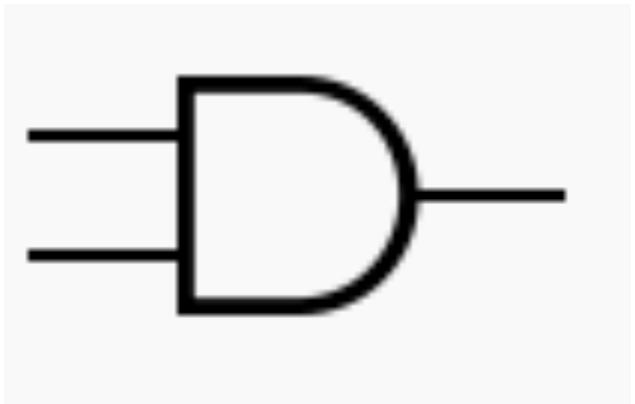
Multiple buffer(3)



The first buffer read-out finished

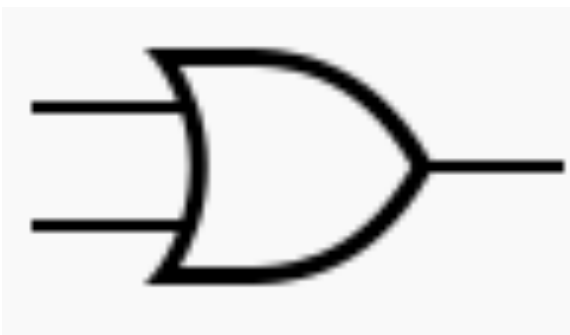
Busy handling

BUSY0
BUSY1
BUSY2



ALL RAM full --> DAQ in stop

BUSY0
BUSY1
BUSY2



NOT EMPTY RAM --> read out

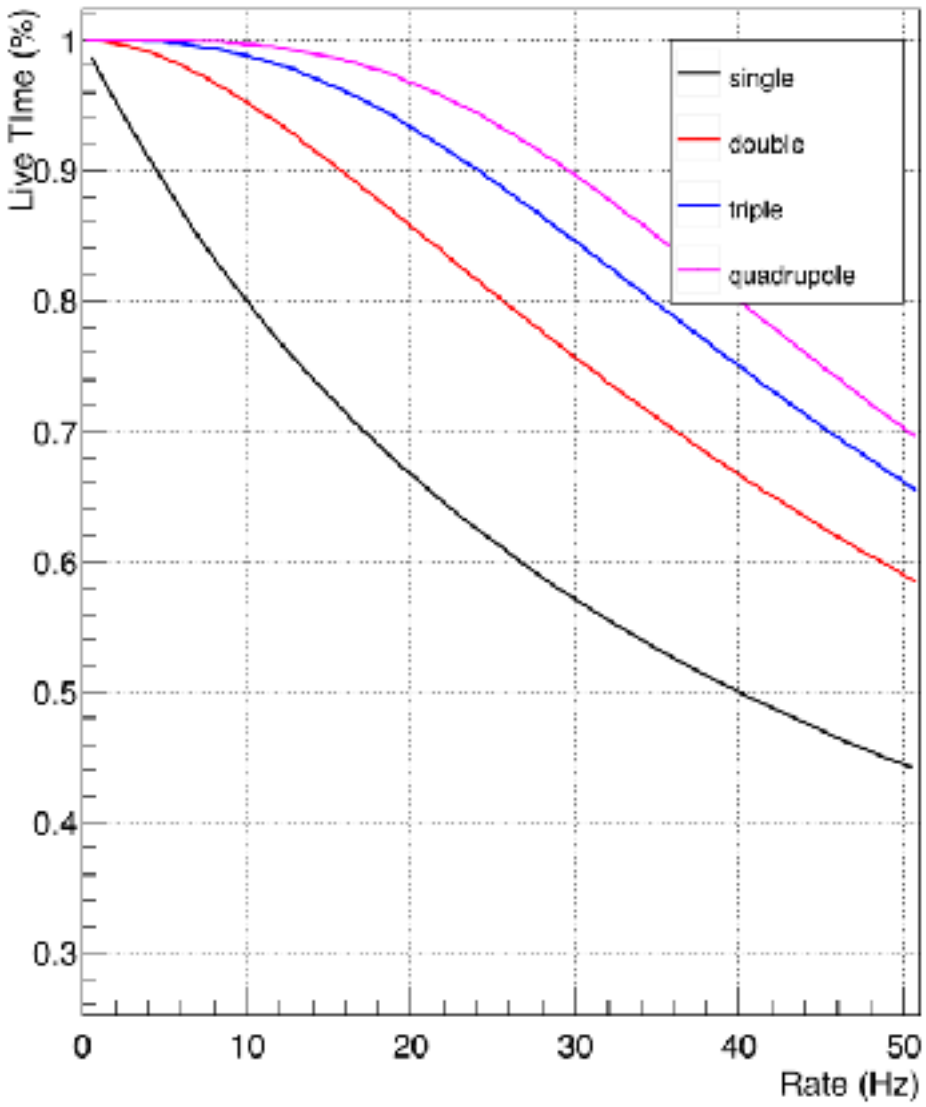
- Read-out when one of the buffer busy is 1
- The system is busy when all the 3 buffers are full

And DAQ Live Time?

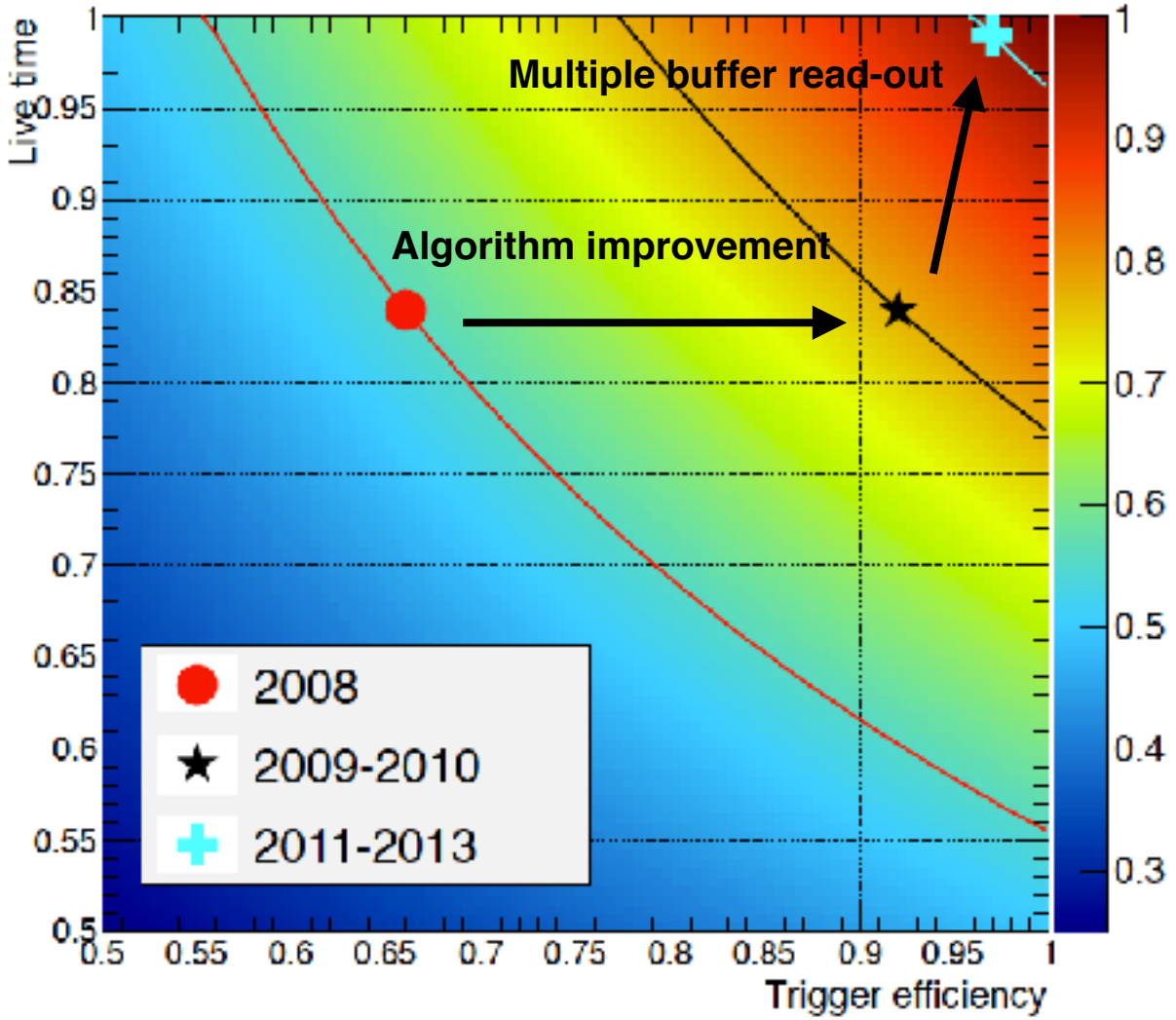
- The system is busy when all the 3 buffers are filled during while you are reading
  - *the read out time is unchanged*
    - ~25ms
  - *the LiveTime fraction can be evaluated*

$$LT = \frac{1 - (t_d \cdot R_{trg})^n}{1 - (t_d \cdot R_{trg})^{n+1}}$$

- it is close to 99% even with a trigger rate close to ~10 Hz
  - *there is room to relax the trigger condition and improve also the trigger efficiency!*



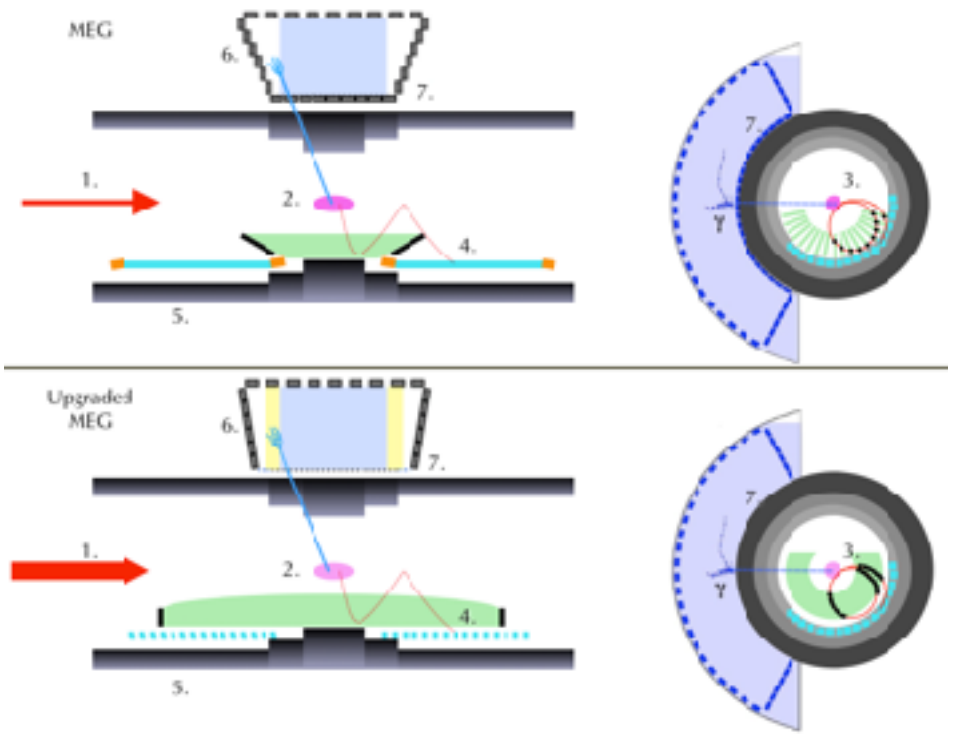
Final DAQ efficiency



Conclusions - part 1

- A TDAQ system to be designed on top of an experiment
  - *the experimental needs drive the choice of the technology*
- Even when the technology is decided a lot of compromise, in this case
  - *Waveform digitiser with lat least 1.6 GSPS sampling speed*
    - background rejection requires the best timing possible
    - trigger latency to be least than 500 ns
  - *trigger based FPGA*
    - system flexibility, for example detector calibration
      - *calibration procedures may changes during the run… and you do want to be the reason why a new calibration procedure will not be used ;)*
- Once your TDAQ is built you have to get the best performance
  - *use all the accessible handles*
    - this is just an example but can help
- **What happens in case of a experiment upgrade?**
  - *let's study the implications of the upgrade*

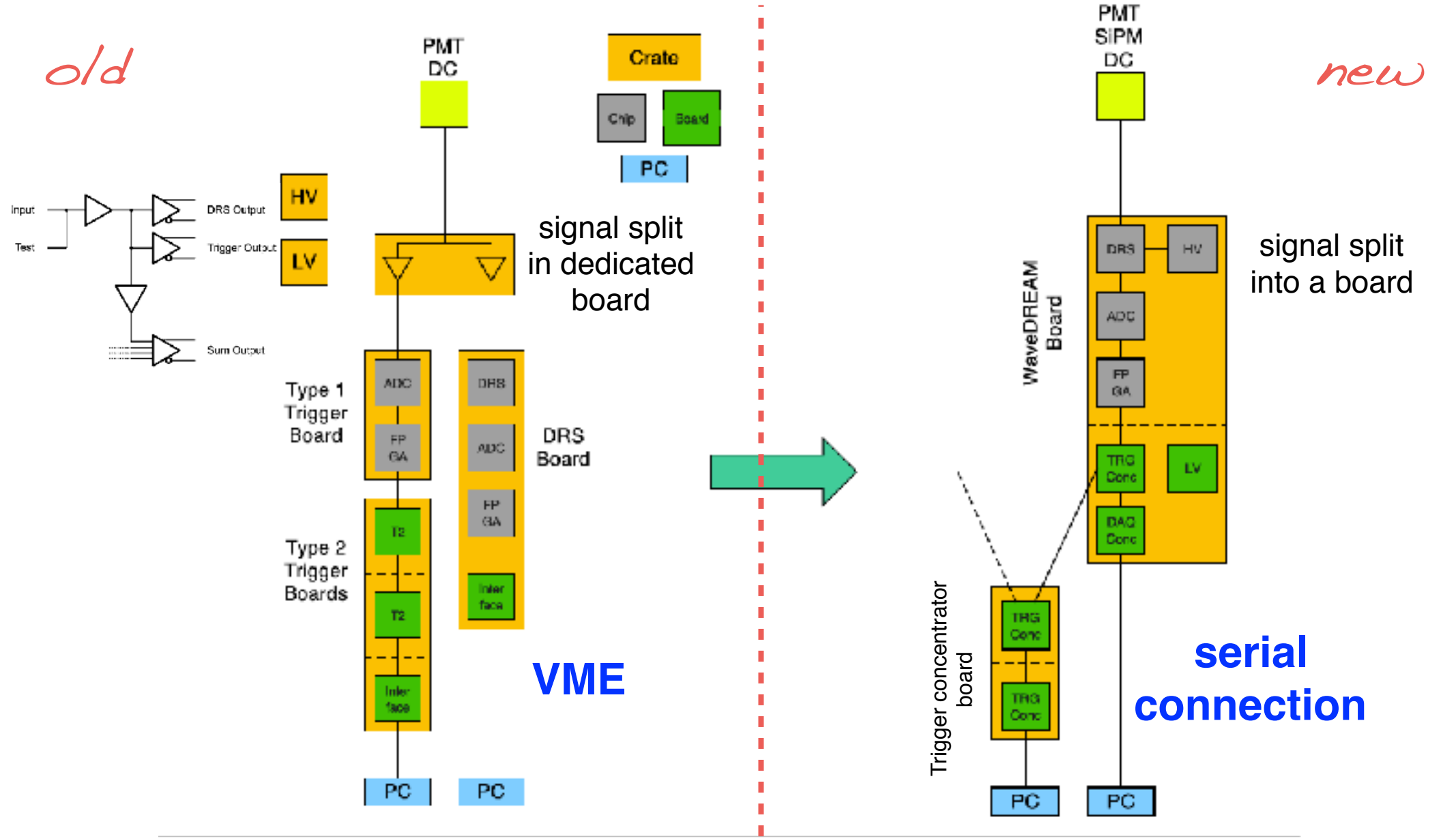
The MEG II experiment



1. **Increasing μ^+-stop** on target
2. **Reducing** target **thickness** to minimize e^+ MS & bremsstrahlung **(or** replace it with an **active target)**
3. **Replacing** the e^+ **tracker** reducing its radiation length and **improving** its **granularity** and **resolution**
4. **Improving** the **timing counter granularity** for **better timing** and **reconstruction**
5. **Improving** the e^+ **tracking-timing integration** by measuring the e^+ trajectory up to the TC interface
6. **Extending γ-ray** detector **acceptance**
7. **Improving** the **γ-ray energy** and **position resolution** for **shallow** events
8. **Integrating splitter, trigger and DAQ** maintaining high bandwidth

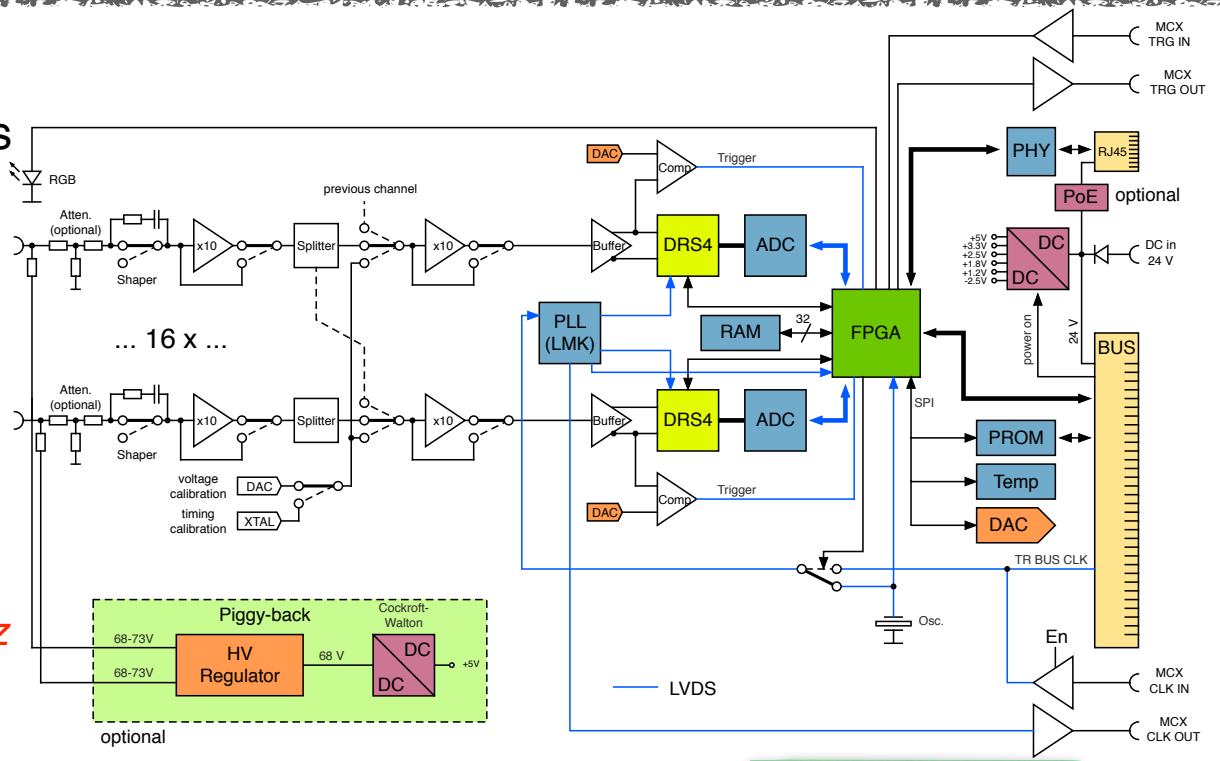
- Number of channels increased by almost a factor of 4
  - *high density TDAQ to fit in the experimental area*
- Doubled beam rate
  - *trigger rate increased by a factor four (~40 Hz)*
    - VME read out not not enough to sustain that value

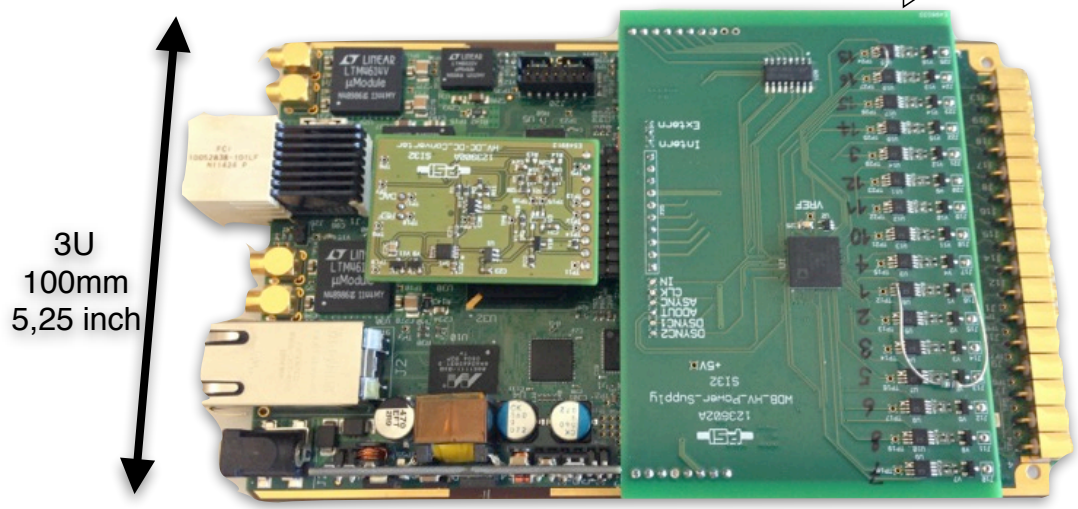
The MEG II TDAQ: innovative solution



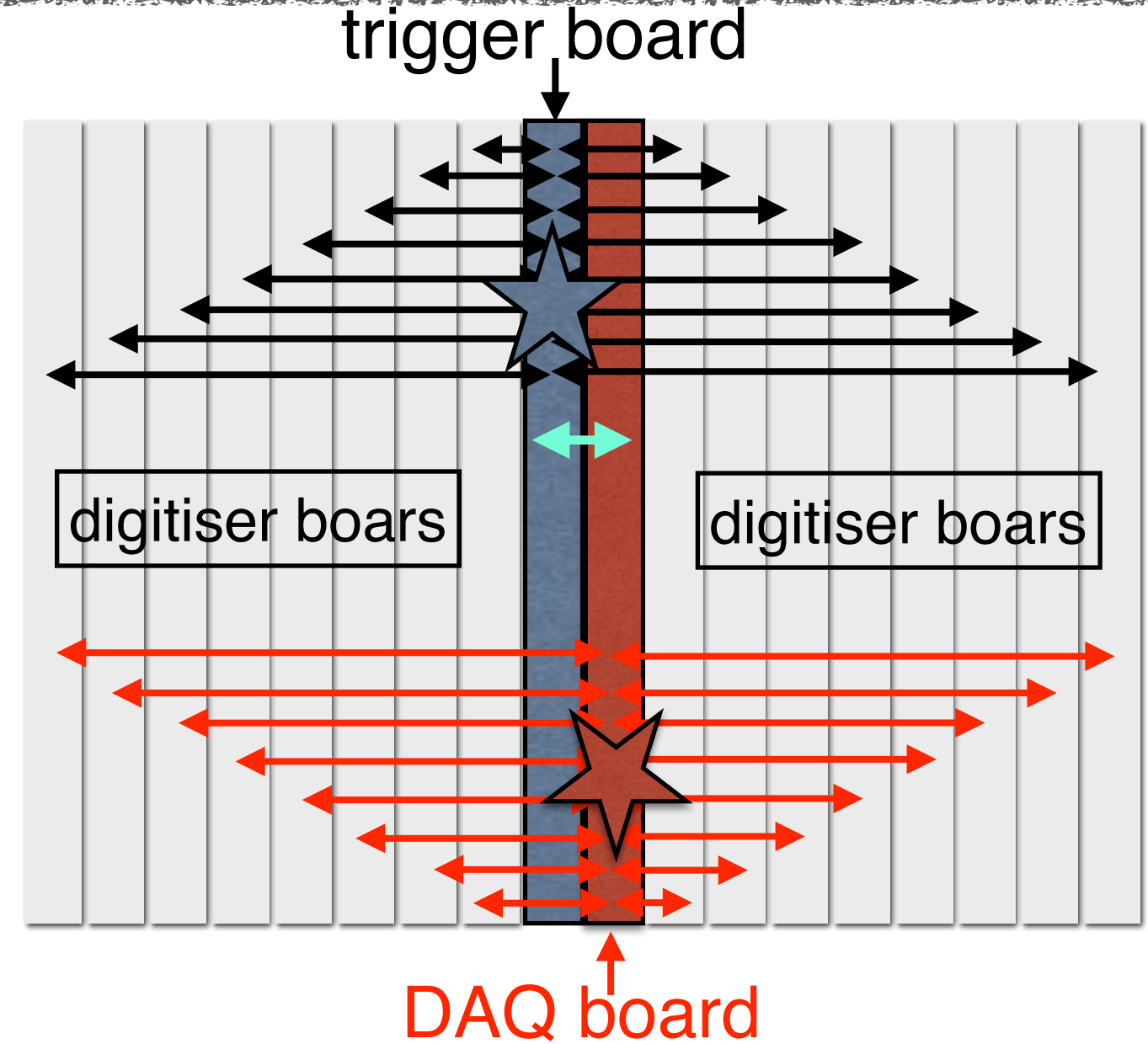
TDAQ integrated board

- Shaper-amplifier circuit for input signals
- DRS4 in bypass mode to forward signals to ADC for trigger
  - *sampling speed = 100 MHz*
  - *Bandwidth on signals = 50 MHz*
    - this follow the Nyquist prescription
- Data processing by FPGA
  - *programmable memory size for 100MHz sampled waveforms*
- SiPM biasing with dedicated and programmable HV regulators
- Stand alone use through ethernet connection and PoE
  - *16 channels TDAQ system*
- 8Gb/s serial connection to TCB (trigger) and 2 Gb/s to DCB (memory read out)





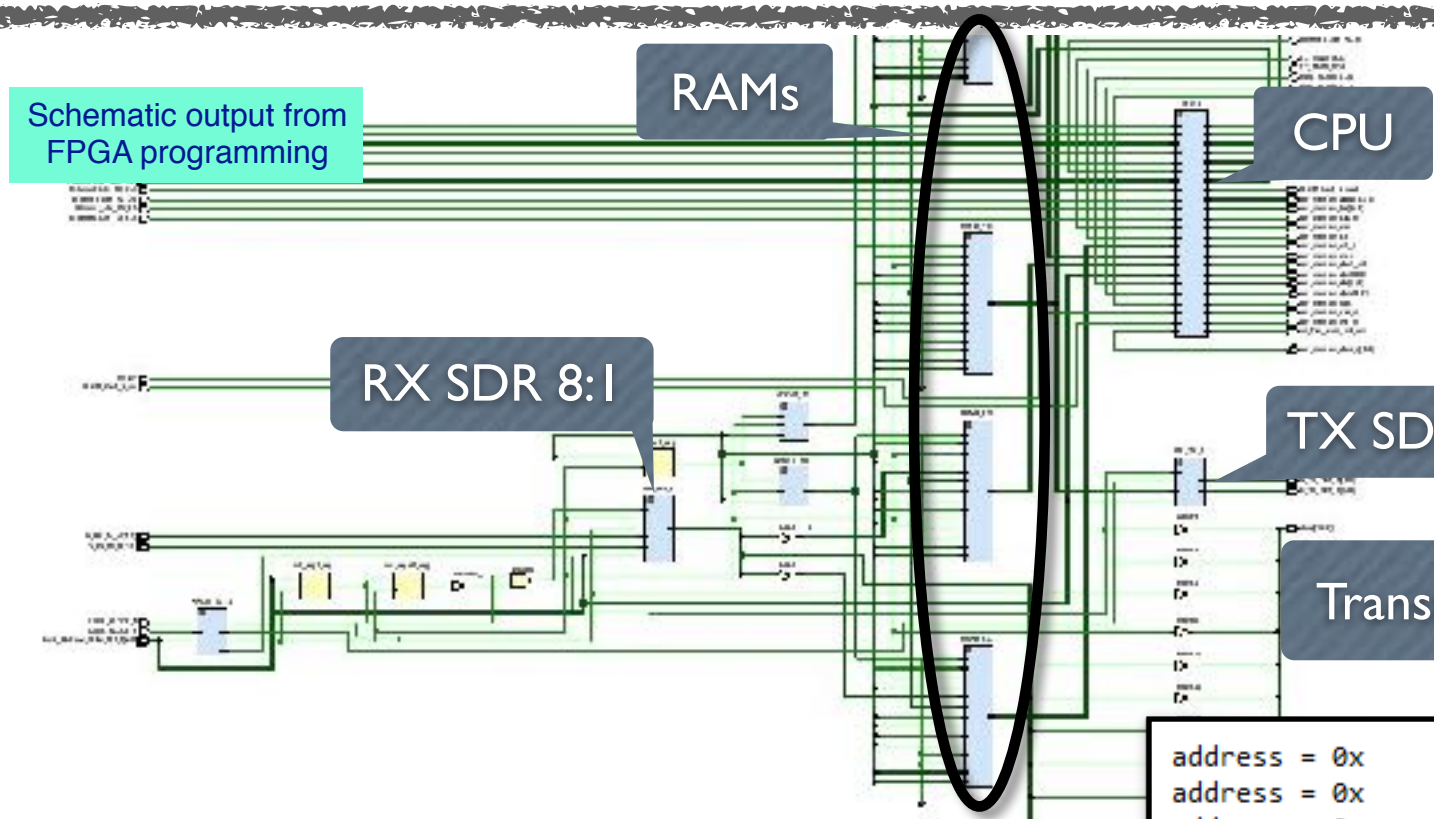
Serial connections



2-STAR topology

Gbit/s read out links embedded in FPGA fabric

R&D on HW: check performance!



Transmission check with SDK

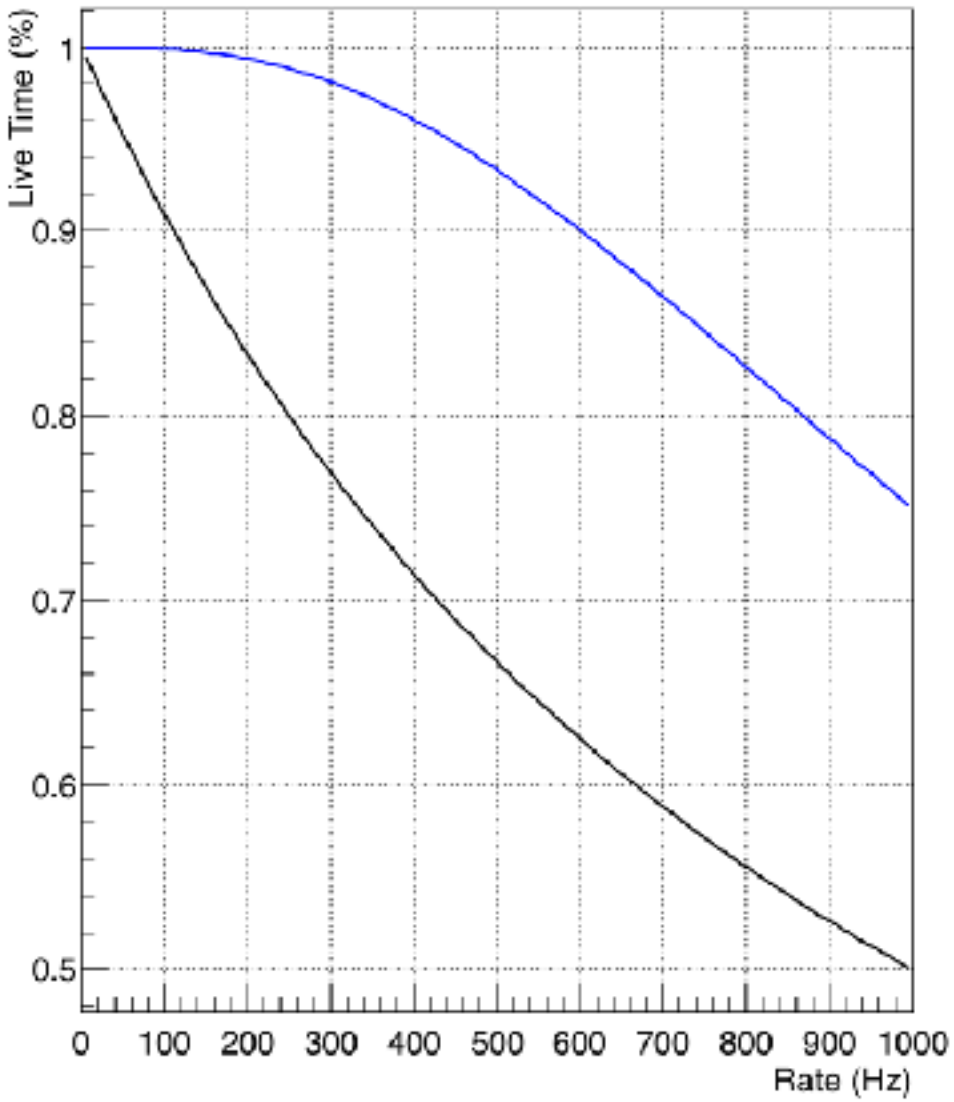
4 ticks latency

```
address = 0x      0, data_tx = 0x   0, data_rx = 0x  1C
address = 0x      1, data_tx = 0x   1, data_rx = 0x  1D
address = 0x      2, data_tx = 0x   2, data_rx = 0x  1E
address = 0x      3, data_tx = 0x   3, data_rx = 0x  1F
address = 0x      4, data_tx = 0x   4, data_rx = 0x   0
address = 0x      5, data_tx = 0x   5, data_rx = 0x   1
address = 0x      6, data_tx = 0x   6, data_rx = 0x   2
address = 0x      7, data_tx = 0x   7, data_rx = 0x   3
address = 0x      8, data_tx = 0x   8, data_rx = 0x   4
address = 0x      9, data_tx = 0x   9, data_rx = 0x   5
address = 0x      A, data_tx = 0x   A, data_rx = 0x   6
address = 0x      B, data_tx = 0x   B, data_rx = 0x   7
address = 0x      C, data_tx = 0x   C, data_rx = 0x   8
address = 0x      D, data_tx = 0x   D, data_rx = 0x   9
address = 0x      E, data_tx = 0x   E, data_rx = 0x   A
```

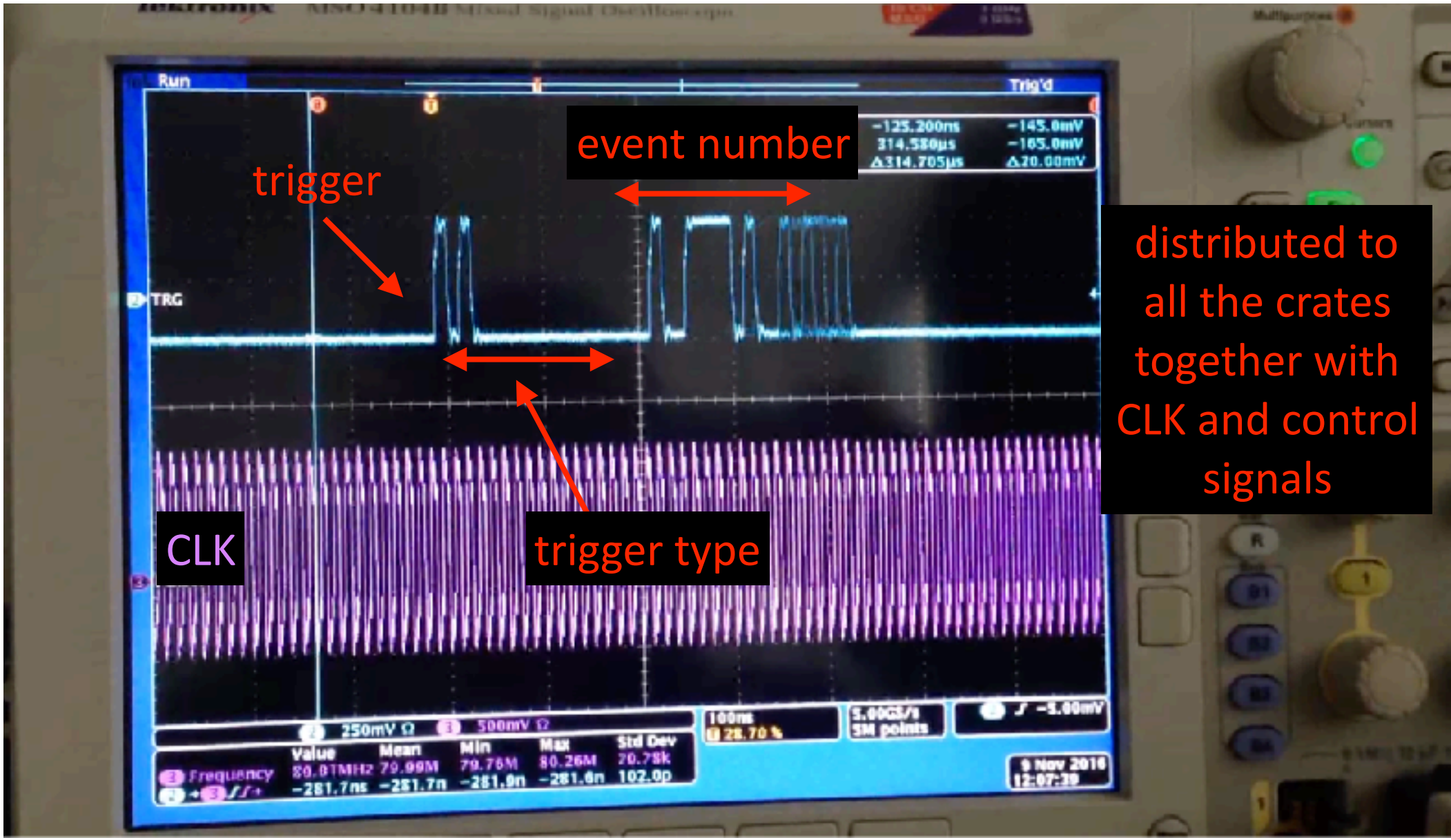
- OUTPUT memory to send out serial data through the cable a then written on INPUT memories
- Synchronised counters to handle memory addresses
- MicroBlaze for memory read out and transmission check
  - *use of SDK to program and access the CPU*
    - automatic check over millions of repetitions
    - test performed also with other data patterns

Improvements with Gbit links!

- Event read out:
  - *performed in broadcast from all the WDB to the DCB*
  - *full (board) event size ~60kB -> event read out in ~1 ms (DRS4 conversion time included)*
    - a factor 25 smaller than in MEG
    - the data transmission is not the bottleneck until ~1 kHz
      - *well above the MEGII foreseen rate*
      - *much faster calibrations!*
      - *We moved the bottleneck to storage capability…*
        - *we are working then for a second level trigger*



Trigger bus 2.0



Conclusions

- A TDAQ system to be designed on top of an experiment
  - *the experimental needs drive the choice of the technology*
- Even when the technology is decided a lot of compromise, in this case
  - *Waveform digitiser with lat least 1.6 GSPS sampling speed*
    - background rejection requires the best timing possible
    - trigger latency to be least than 500 ns
  - *trigger based FPGA*
    - system flexibility, for example detector calibration
      - *calibration procedures may changes during the run… and you do want to be the reason why a new calibration procedure will not be used ;)*
- Once your TDAQ is built you have to get the best performance
  - *use all the accessible handles*
    - this is just an example but can help
- The new TDAQ system for the MEG II experiment
  - *keep the good features of the previous design and point to the critical points*
    - Trigger and DAQ merged in one (custom) board
      - supported by a concentrating trigger crate
    - new serial data transmission scheme to speed up event read out and transmission inside the trigger system
    - improve timing algorithm to improve background rejection